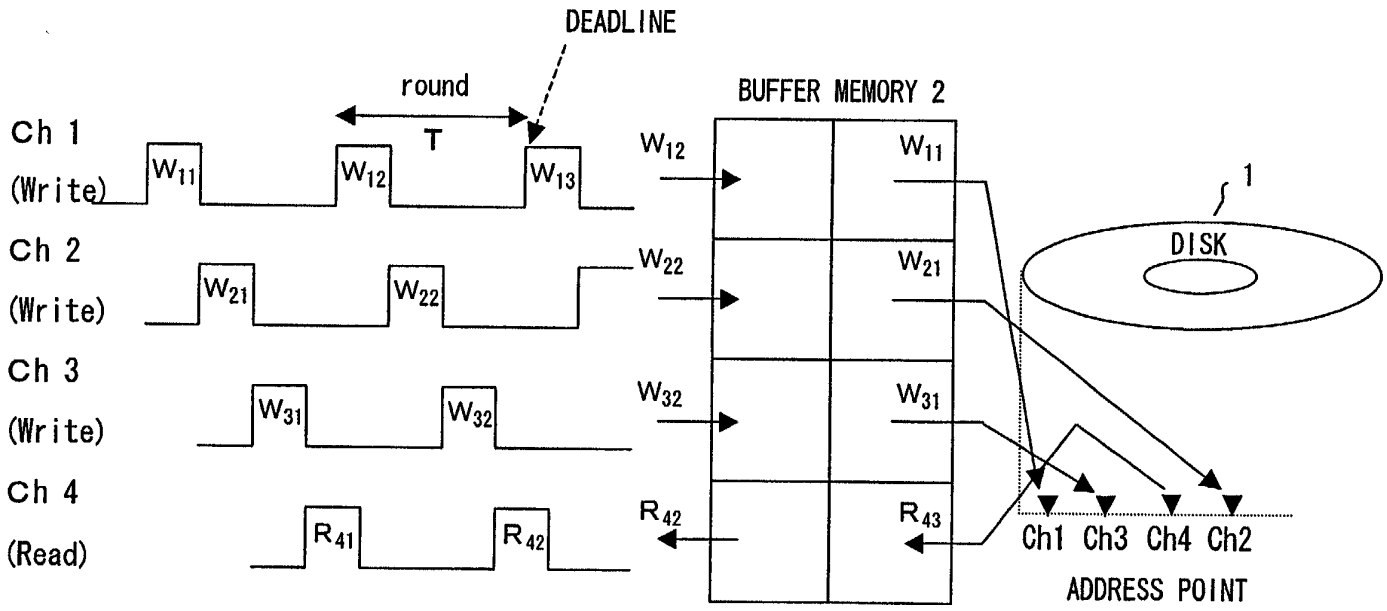


FIG. 1

PRIOR ART

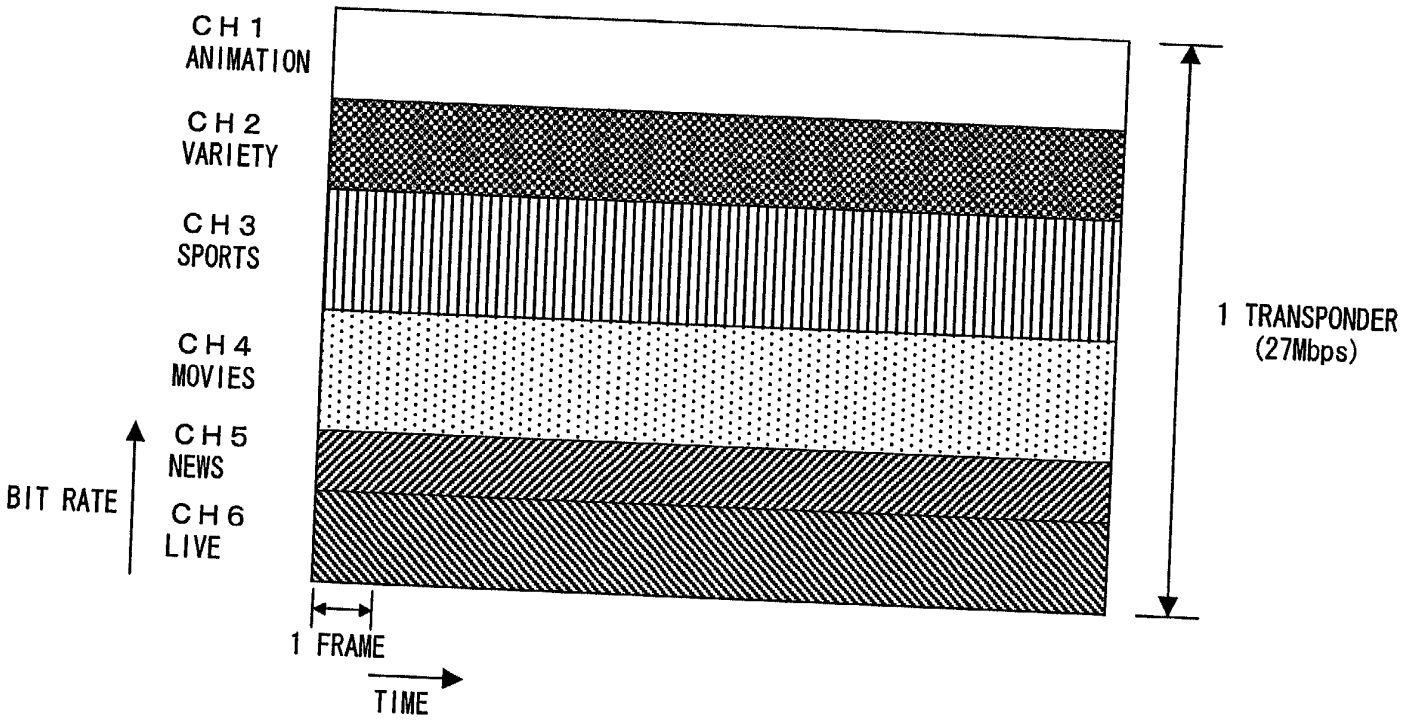

CH 1
ANIMATION
CH 2
VARIETY
CH 3
SPORTS
CH 4
MOVIES
CH 5
NEWS
CH 6
LIVE
BIT RATE
1 FRAME
TIME
1 TRANSPONDER
(27Mbps)

FIG. 2

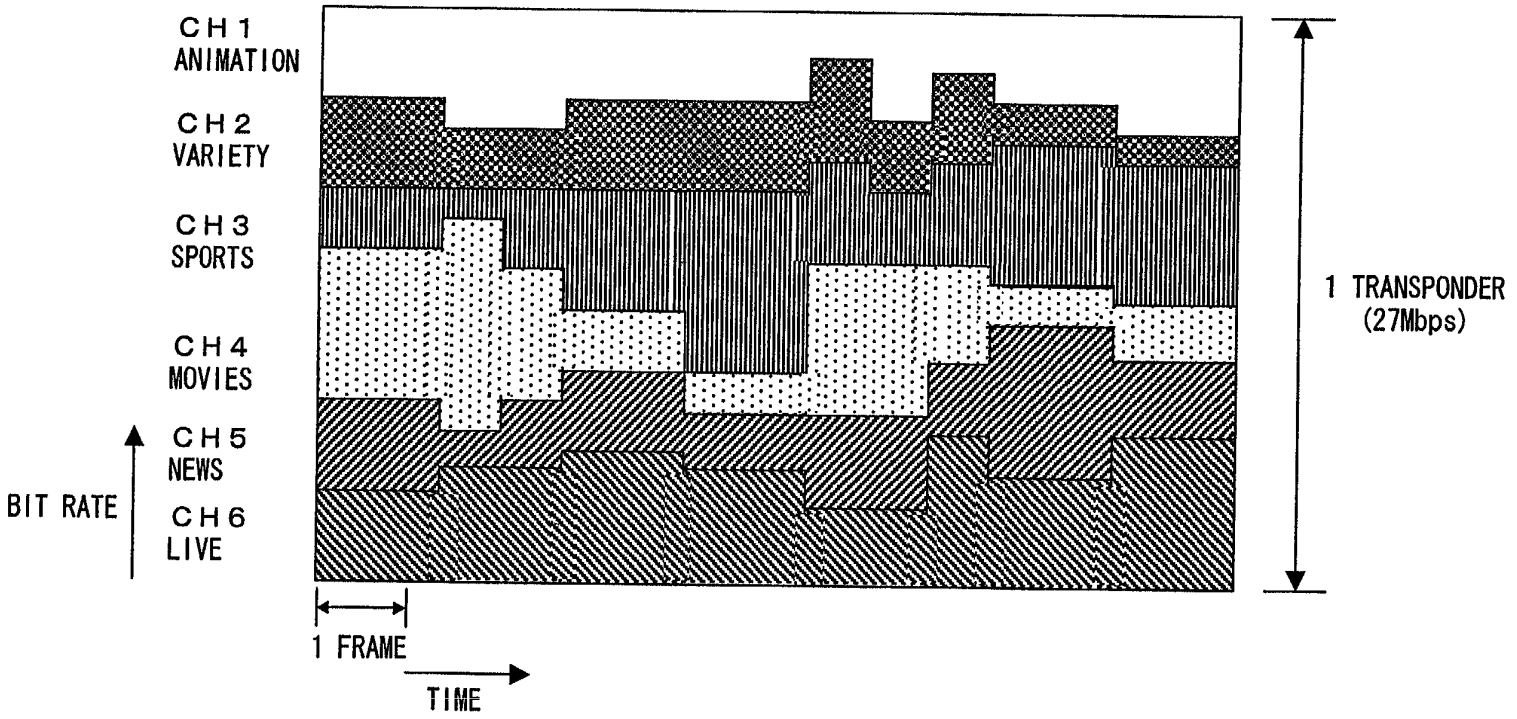

CH 1
ANIMATION
CH 2
VARIETY
CH 3
SPORTS
CH 4
MOVIES
CH 5
NEWS
CH 6
LIVE
BIT RATE
1 FRAME
TIME
1 TRANSPONDER
(27Mbps)

FIG. 3

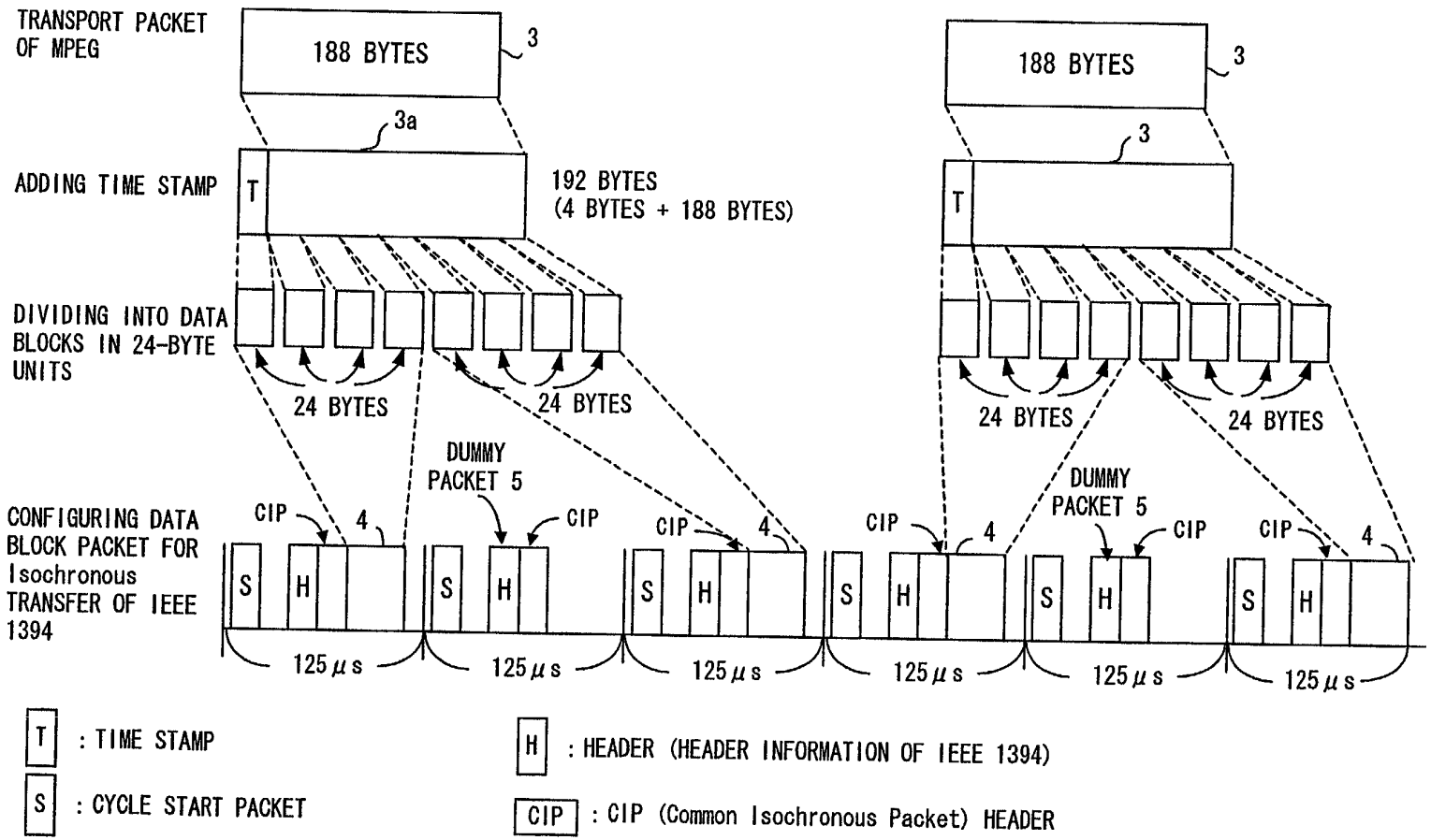

TRANSPORT PACKET OF MPEG
188 BYTES
3
3a
ADDING TIME STAMP
T
192 BYTES
(4 BYTES + 188 BYTES)
DIVIDING INTO DATA BLOCKS IN 24-BYTE UNITS
24 BYTES
24 BYTES
DUMMY PACKET 5
CONFIGURING DATA BLOCK PACKET FOR Isochronous TRANSFER OF IEEE 1394
CIP
4
S
H
125 μs
T : TIME STAMP
S : CYCLE START PACKET
H : HEADER (HEADER INFORMATION OF IEEE 1394)
CIP : CIP (Common Isochronous Packet) HEADER

FIG. 4

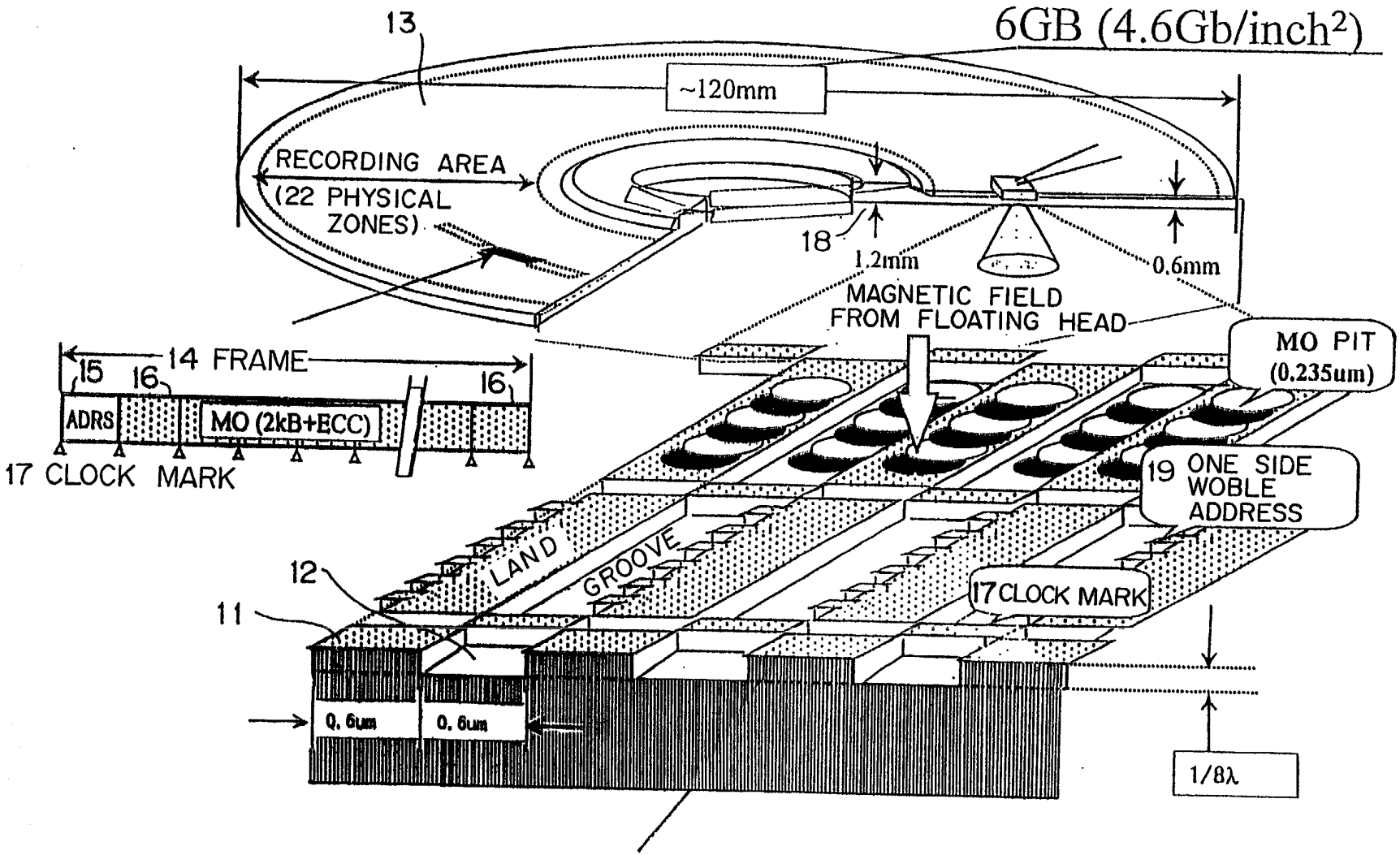

6GB (4.6Gb/inch²)
13
~120mm
RECORDING AREA
(22 PHYSICAL ZONES)
18
1.2mm
0.6mm
MAGNETIC FIELD FROM FLOATING HEAD
MO PIT (0.235um)
14 FRAME
15
16
16
ADRS
MO (2kB+ECC)
17 CLOCK MARK
19 ONE SIDE WOBLE ADDRESS
LAND
GROOVE
12
11
17CLOCK MARK
0.6um
0.6um
1/8λ

FIG. 5

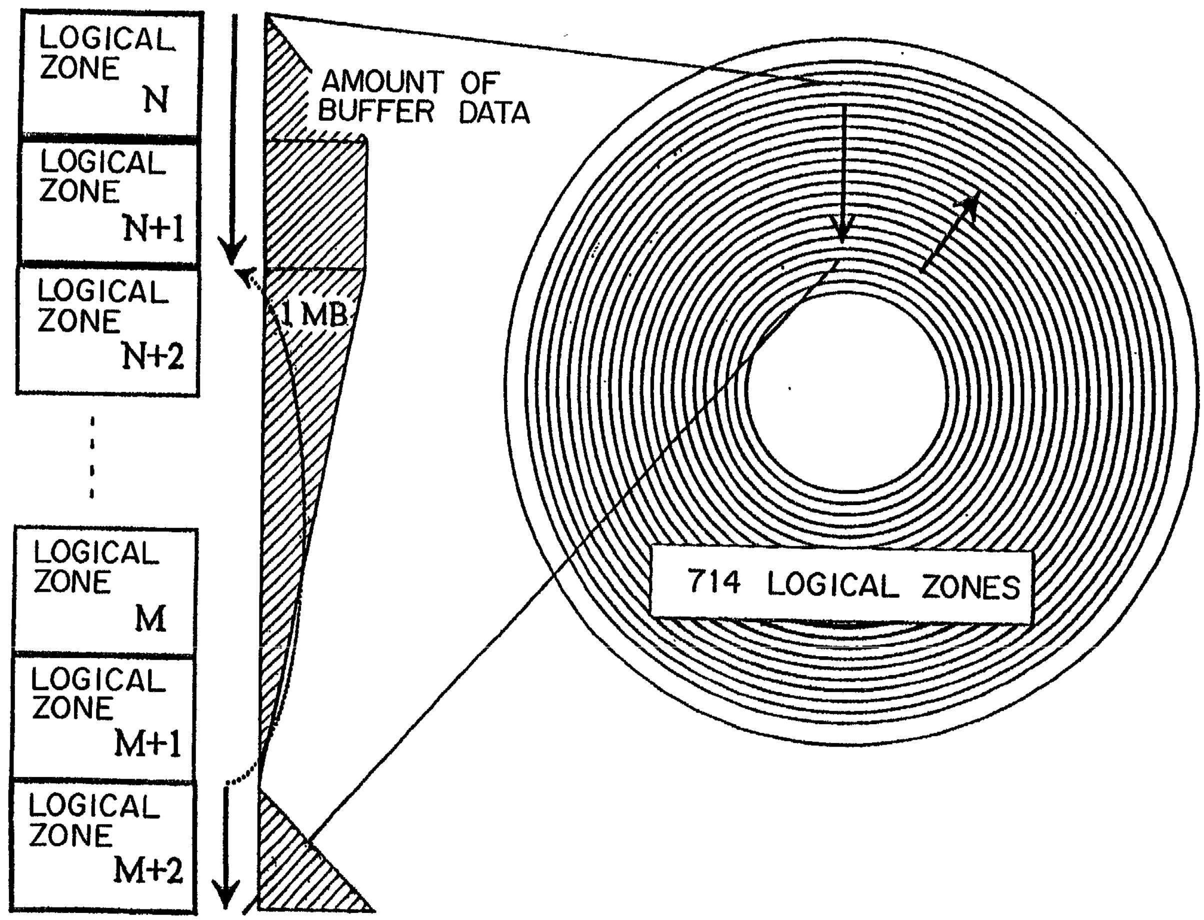

LOGICAL ZONE N
LOGICAL ZONE N+1
LOGICAL ZONE N+2
LOGICAL ZONE M
LOGICAL ZONE M+1
LOGICAL ZONE M+2
AMOUNT OF BUFFER DATA
1 MB
714 LOGICAL ZONES

FIG. 6

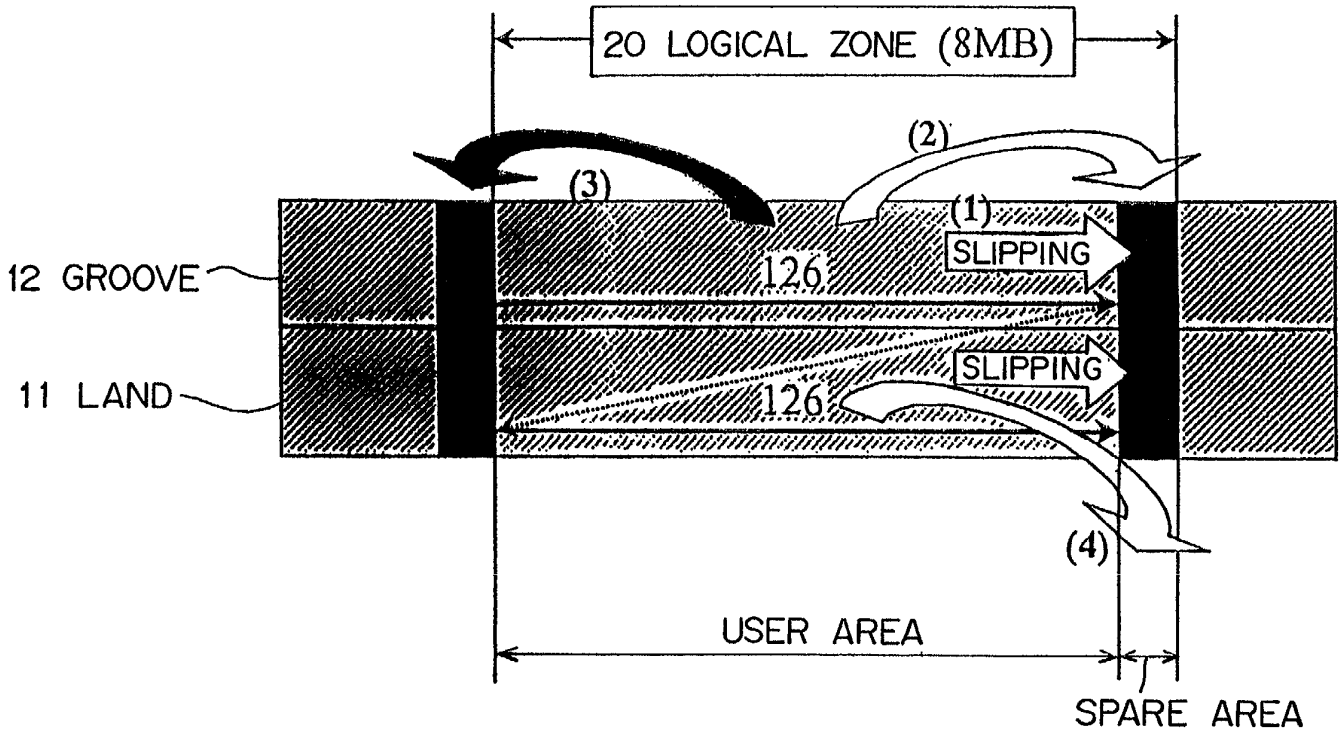

20 LOGICAL ZONE (8MB)
(2)
(3)
(1)
SLIPPING
126
12 GROOVE
11 LAND
SLIPPING
126
(4)
USER AREA
SPARE AREA

FIG. 7

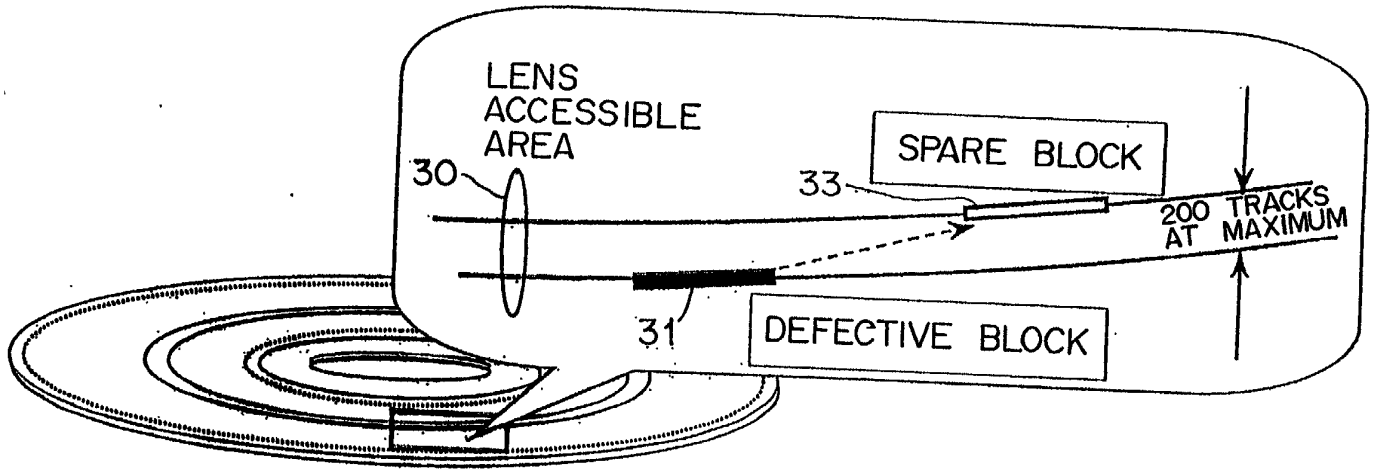

LENS
ACCESSIBLE
AREA
30
33
SPARE BLOCK
200 TRACKS
AT MAXIMUM
31
DEFECTIVE BLOCK

FIG. 8

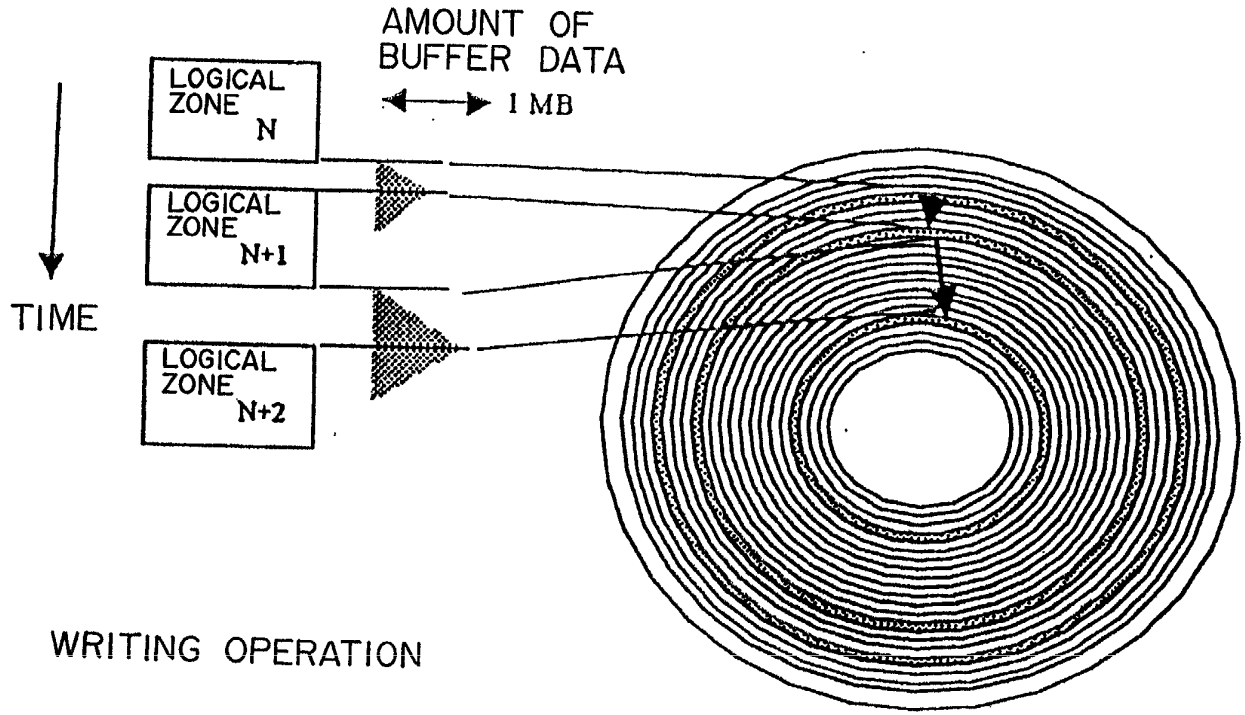

AMOUNT OF
BUFFER DATA
1 MB
LOGICAL ZONE N
LOGICAL ZONE N+1
LOGICAL ZONE N+2
TIME
WRITING OPERATION

FIG. 9

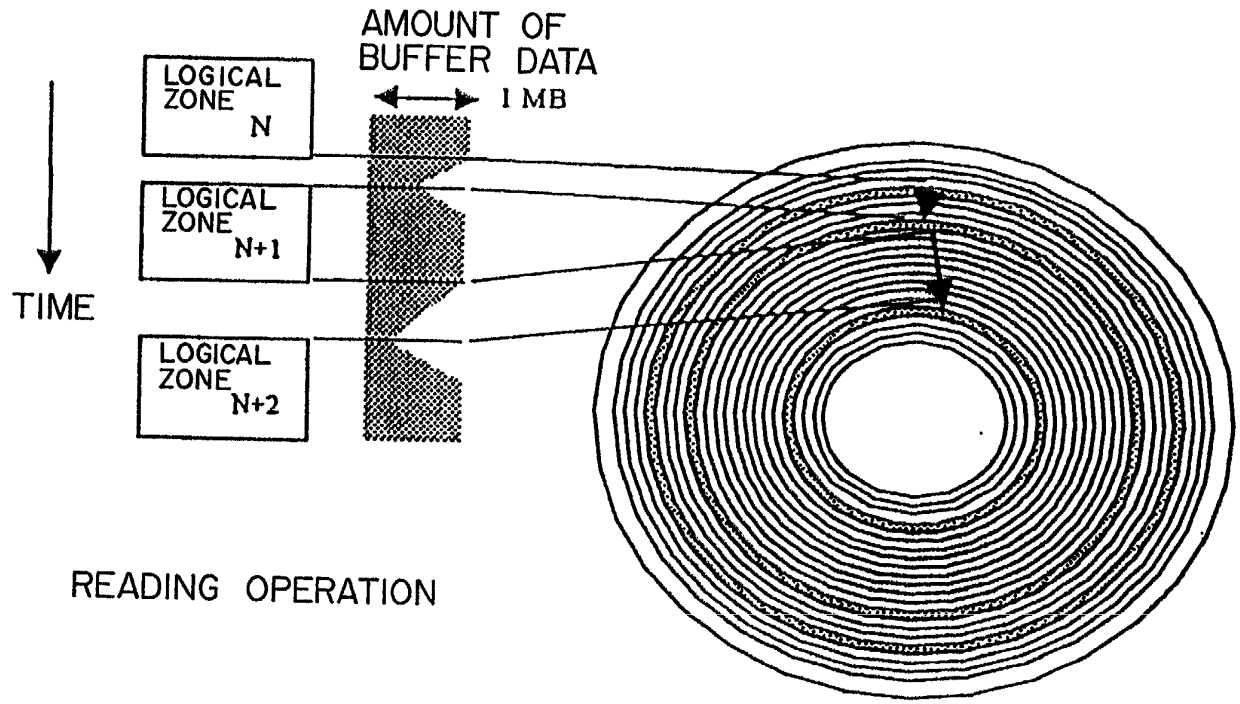

AMOUNT OF
BUFFER DATA
1 MB
LOGICAL
ZONE
N
LOGICAL
ZONE
N+1
LOGICAL
ZONE
N+2
TIME
READING OPERATION

FIG. 10

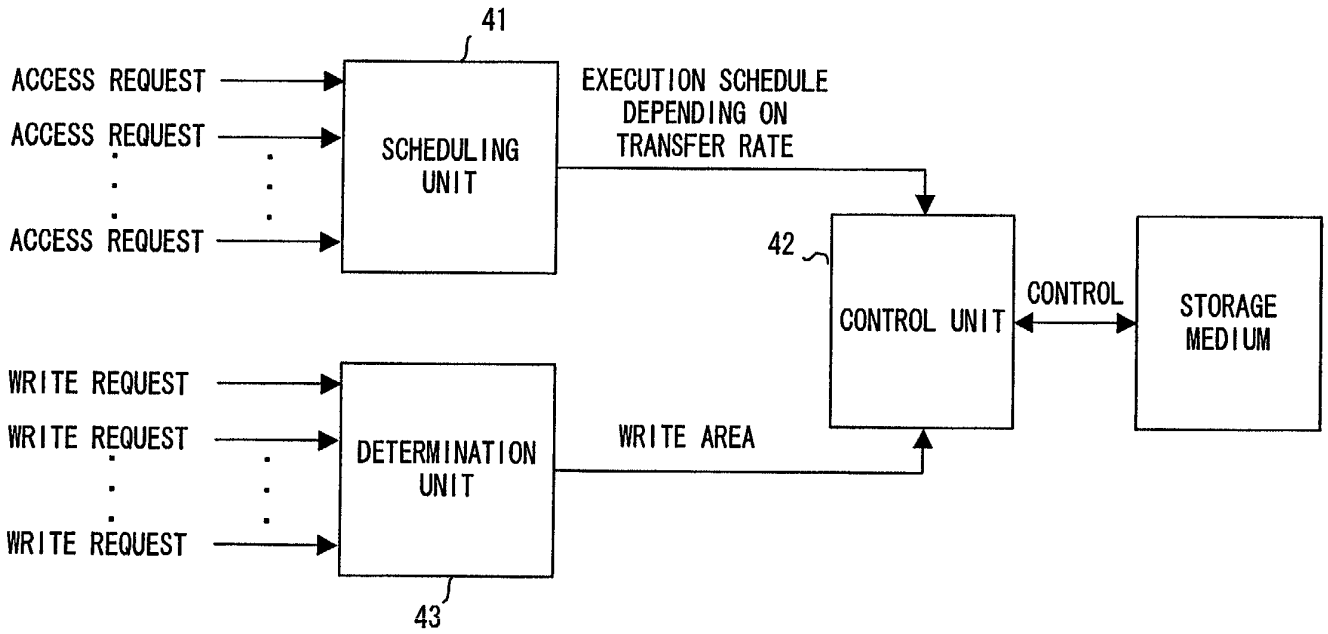

41
ACCESS REQUEST
ACCESS REQUEST
ACCESS REQUEST
SCHEDULING UNIT
EXECUTION SCHEDULE DEPENDING ON TRANSFER RATE
42
CONTROL UNIT
CONTROL
STORAGE MEDIUM
WRITE REQUEST
WRITE REQUEST
WRITE REQUEST
DETERMINATION UNIT
WRITE AREA
43

FIG. 11

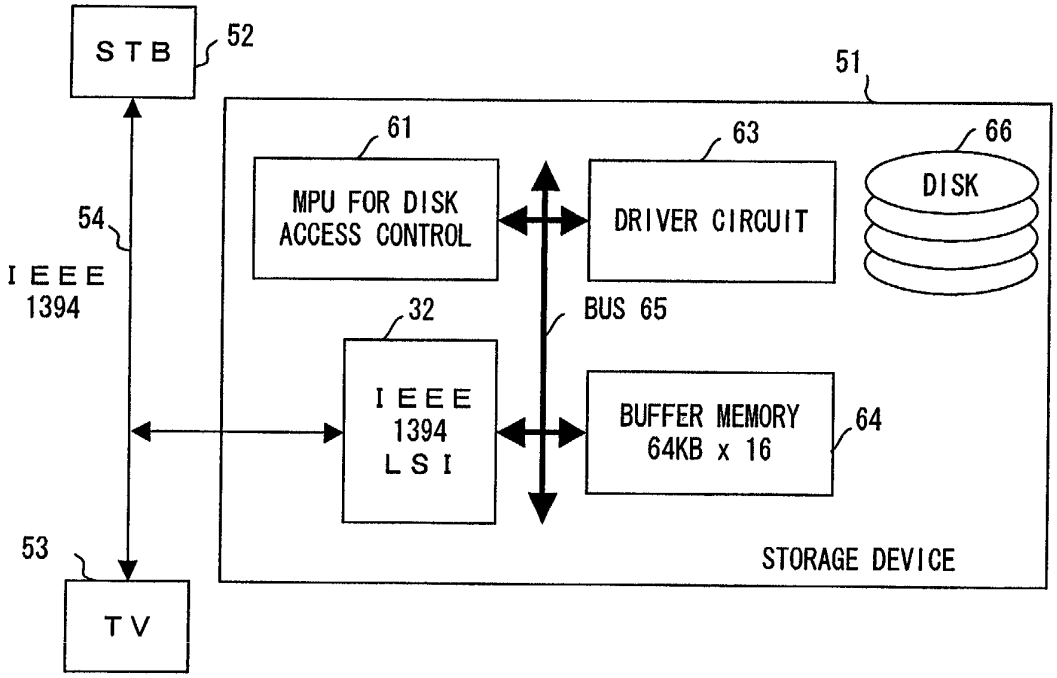

STB
52
51
61
MPU FOR DISK
ACCESS CONTROL
63
DRIVER CIRCUIT
66
DISK
54
IEEE
1394
32
BUS 65
IEEE
1394
LSI
BUFFER MEMORY
64KB x 16
64
53
TV
STORAGE DEVICE

FIG. 12

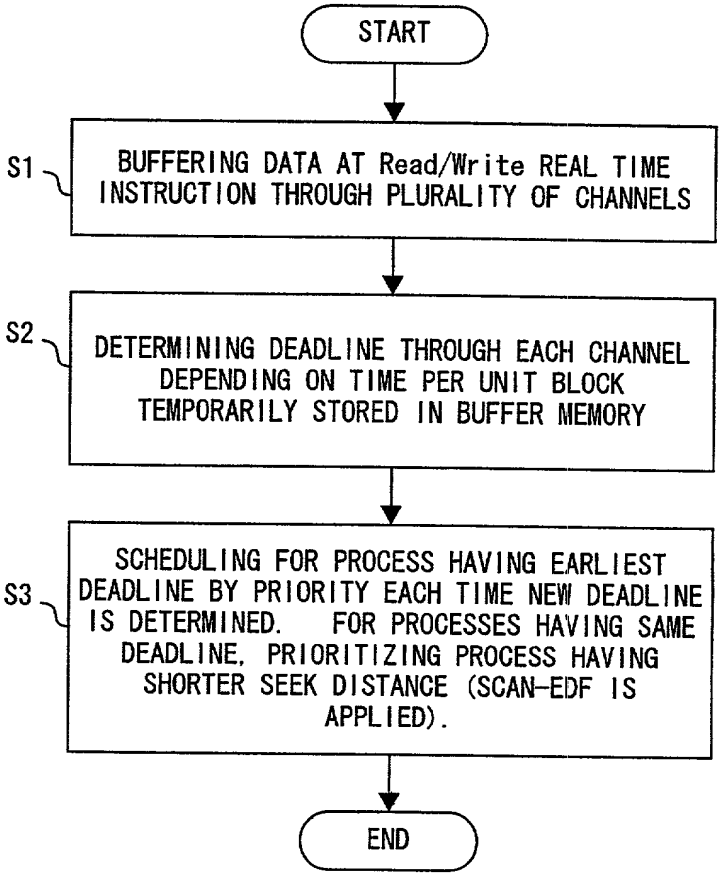

START
S1
BUFFERING DATA AT Read/Write REAL TIME INSTRUCTION THROUGH PLURALITY OF CHANNELS
S2
DETERMINING DEADLINE THROUGH EACH CHANNEL DEPENDING ON TIME PER UNIT BLOCK TEMPORARILY STORED IN BUFFER MEMORY
S3
SCHEDULING FOR PROCESS HAVING EARLIEST DEADLINE BY PRIORITY EACH TIME NEW DEADLINE IS DETERMINED. FOR PROCESSES HAVING SAME DEADLINE, PRIORITIZING PROCESS HAVING SHORTER SEEK DISTANCE (SCAN-EDF IS APPLIED).
END

FIG. 13

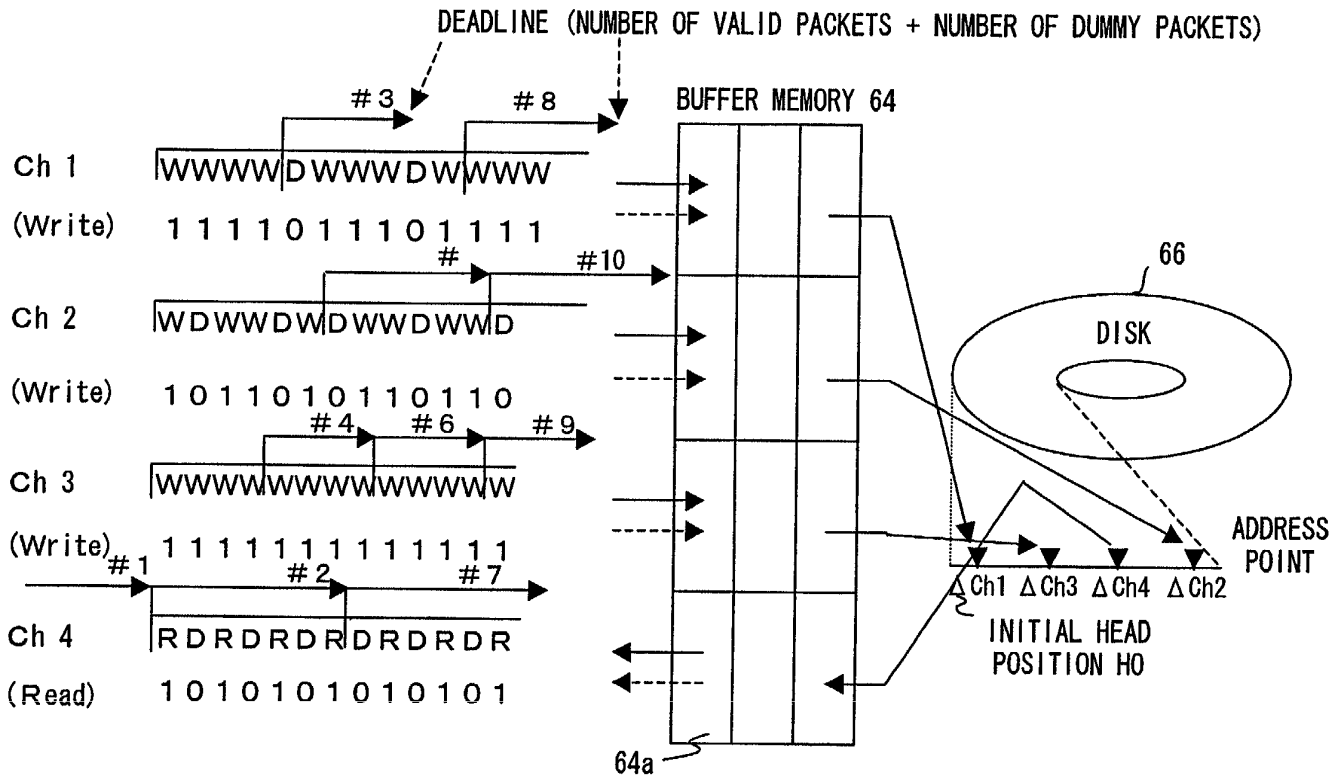
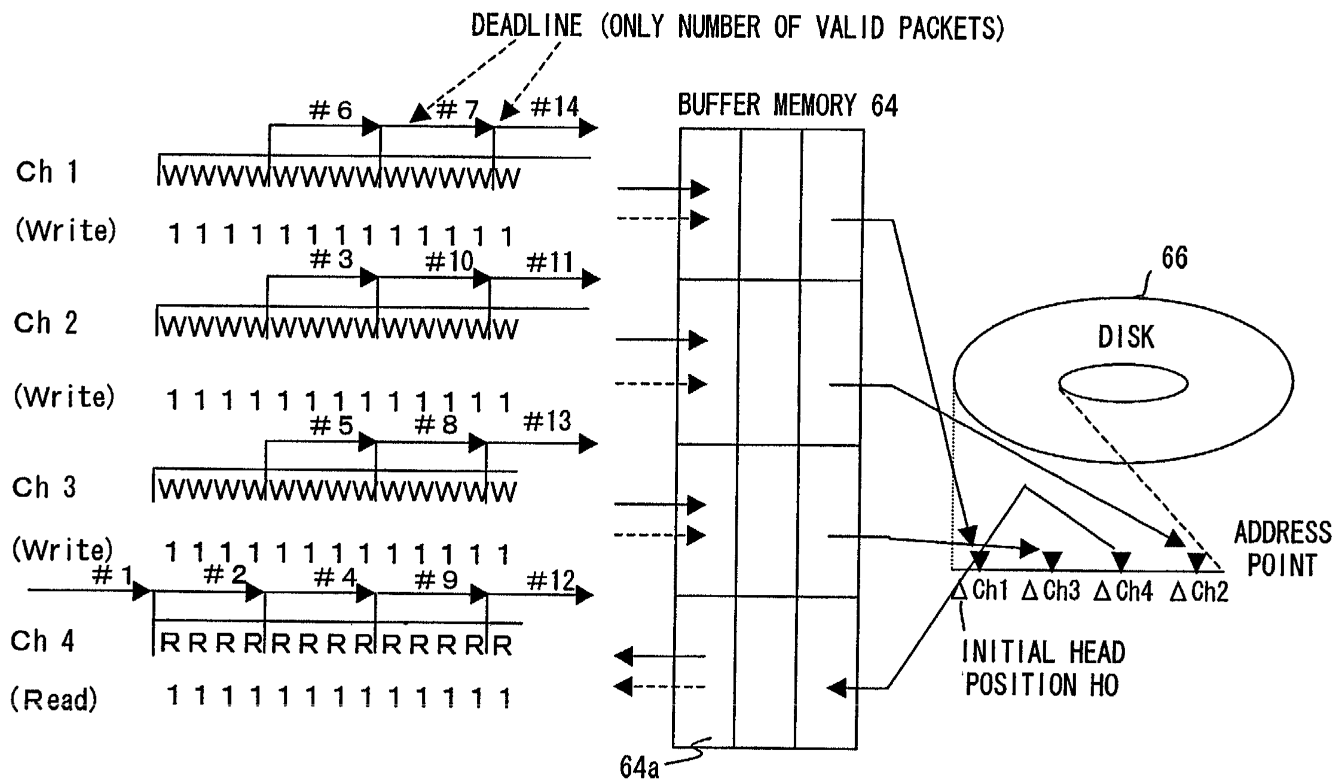

DEADLINE (NUMBER OF VALID PACKETS + NUMBER OF DUMMY PACKETS)
#3
#8
BUFFER MEMORY 64
Ch 1
WWWWDWWWDWWWW
(Write)
1111011101111
#
#10
66
Ch 2
WDWWDWDWWDWWD
DISK
(Write)
1011010110110
#4
#6
#9
Ch 3
WWWWWWWWWWWWW
ADDRESS POINT
(Write)
1111111111111
#1
#2
#7
Ch 4
RDRDRDRDRDRDR
ΔCh1 ΔCh3 ΔCh4 ΔCh2
(Read)
1010101010101
INITIAL HEAD POSITION H0
64a

FIG. 14

MAXIMUM TRANSFER RATE (NUMBER OF BYTES/ NUMBER OF PACKETS)	DEADLINE INFORMATION	BINARY DATA	VALID DATA

FIG. 15


DEADLINE (ONLY NUMBER OF VALID PACKETS)
BUFFER MEMORY 64
Ch 1
(Write)
Ch 2
(Write)
Ch 3
(Write)
Ch 4
(Read)
#1
#2
#3
#4
#5
#6
#7
#8
#9
#10
#11
#12
#13
#14
66
DISK
ADDRESS POINT
ΔCh1 ΔCh3 ΔCh4 ΔCh2
INITIAL HEAD POSITION HO
64a

F I G. 16

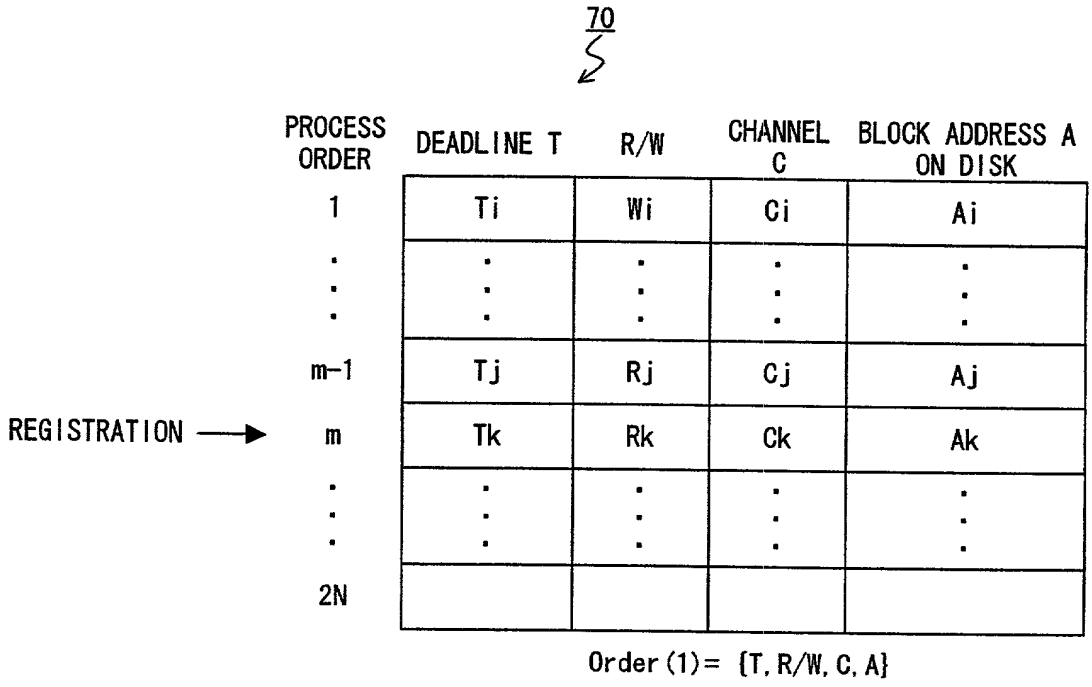

70
PROCESS ORDER
DEADLINE T
R/W
CHANNEL C
BLOCK ADDRESS A ON DISK
1
Ti
Wi
Ci
Ai
m-1
Tj
Rj
Cj
Aj
REGISTRATION
m
Tk
Rk
Ck
Ak
2N
Order (1) = {T, R/W, C, A}

FIG. 17

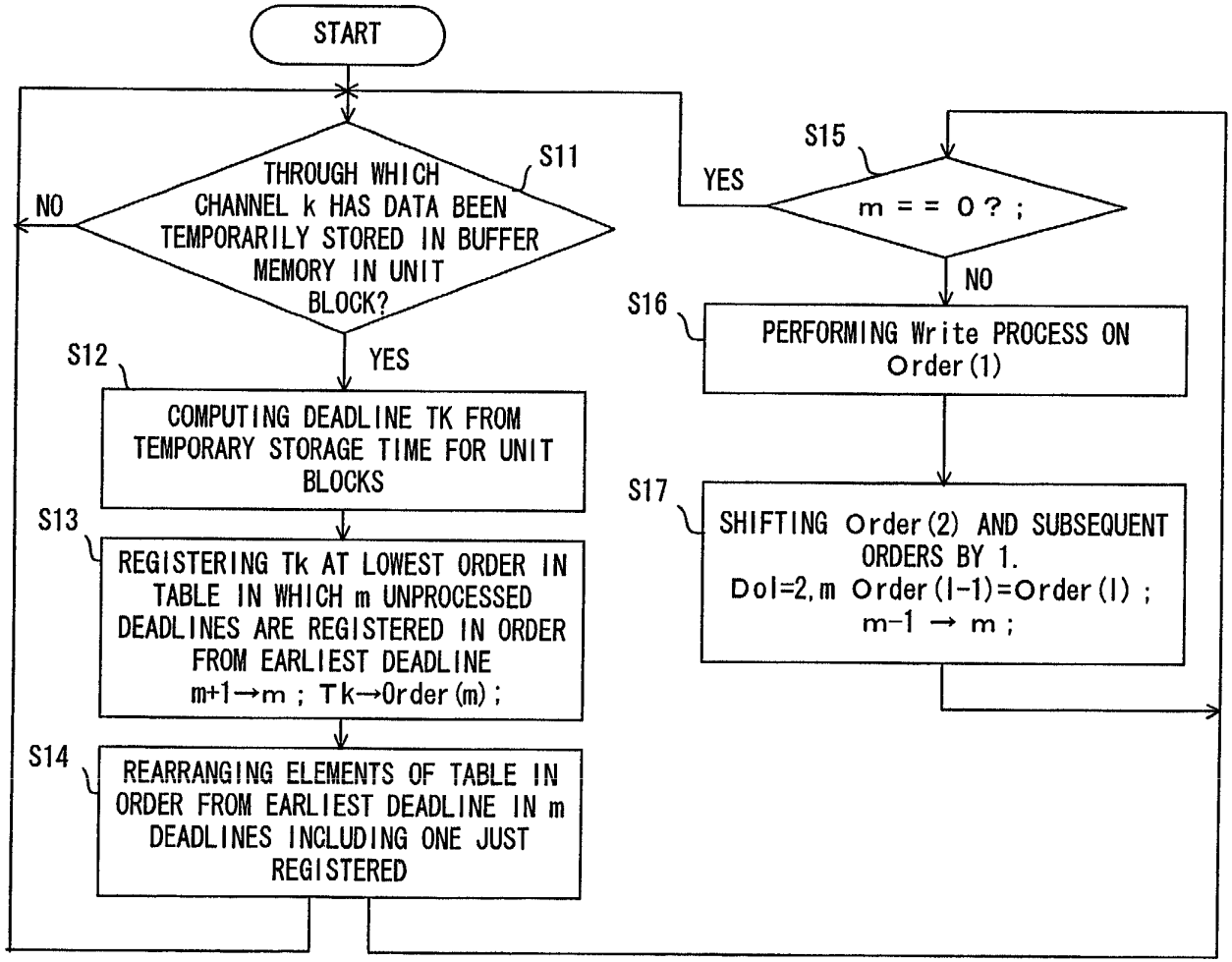

START
S11
THROUGH WHICH CHANNEL k HAS DATA BEEN TEMPORARILY STORED IN BUFFER MEMORY IN UNIT BLOCK?
NO
YES
S12
COMPUTING DEADLINE TK FROM TEMPORARY STORAGE TIME FOR UNIT BLOCKS
S13
REGISTERING Tk AT LOWEST ORDER IN TABLE IN WHICH m UNPROCESSED DEADLINES ARE REGISTERED IN ORDER FROM EARLIEST DEADLINE
m+1→m ; Tk→Order(m);
S14
REARRANGING ELEMENTS OF TABLE IN ORDER FROM EARLIEST DEADLINE IN m DEADLINES INCLUDING ONE JUST REGISTERED
S15
m = = 0 ? ;
YES
NO
S16
PERFORMING Write PROCESS ON Order(1)
S17
SHIFTING Order(2) AND SUBSEQUENT ORDERS BY 1.
Do I=2,m Order(I-1)=Order(I) ;
m-1 → m ;

FIG. 18

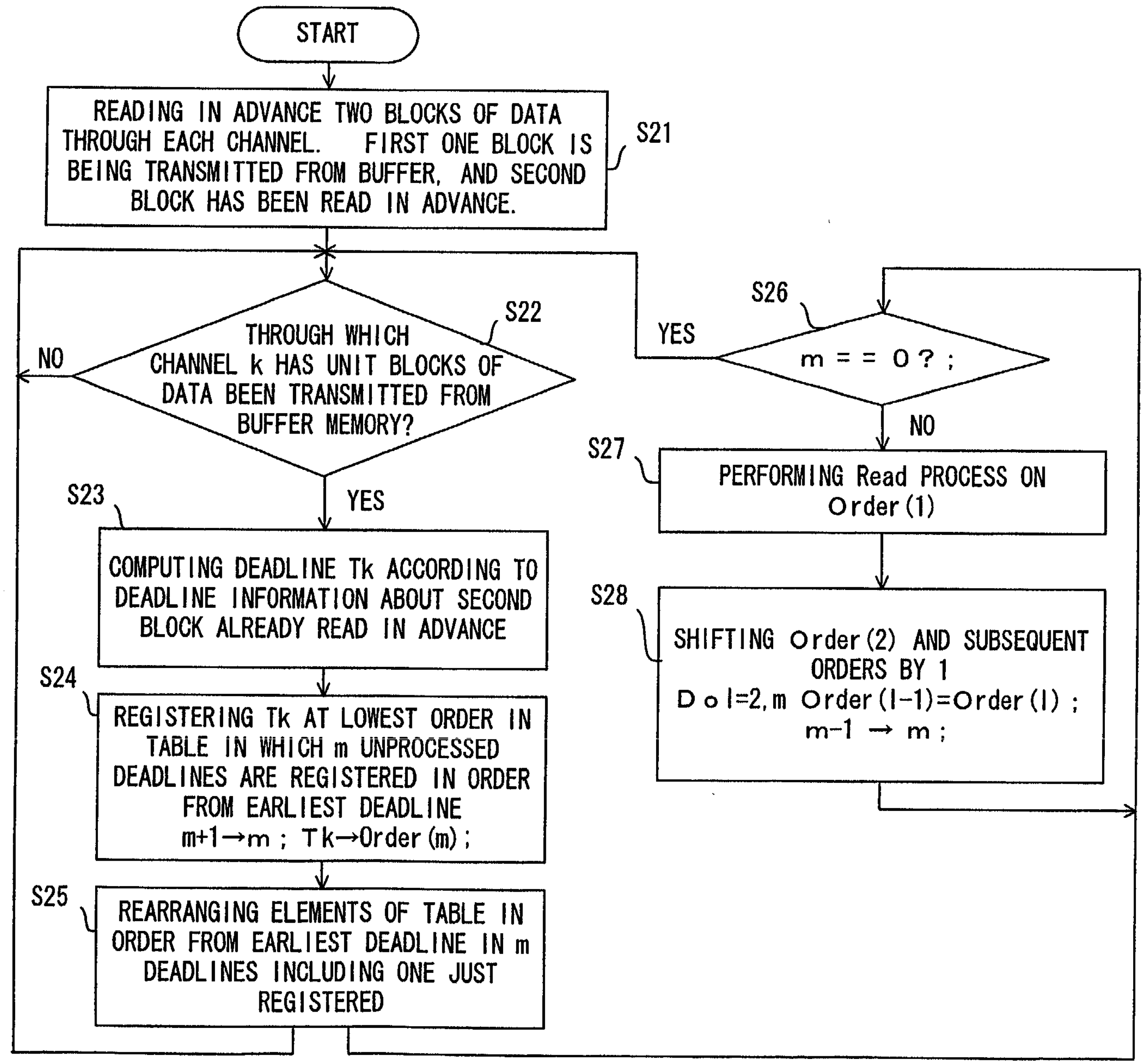

START
READING IN ADVANCE TWO BLOCKS OF DATA THROUGH EACH CHANNEL. FIRST ONE BLOCK IS BEING TRANSMITTED FROM BUFFER, AND SECOND BLOCK HAS BEEN READ IN ADVANCE.
S21
THROUGH WHICH CHANNEL k HAS UNIT BLOCKS OF DATA BEEN TRANSMITTED FROM BUFFER MEMORY?
S22
NO
YES
S23
COMPUTING DEADLINE Tk ACCORDING TO DEADLINE INFORMATION ABOUT SECOND BLOCK ALREADY READ IN ADVANCE
S24
REGISTERING Tk AT LOWEST ORDER IN TABLE IN WHICH m UNPROCESSED DEADLINES ARE REGISTERED IN ORDER FROM EARLIEST DEADLINE
m+1→m ; Tk→Order (m) ;
S25
REARRANGING ELEMENTS OF TABLE IN ORDER FROM EARLIEST DEADLINE IN m DEADLINES INCLUDING ONE JUST REGISTERED
S26
YES
m = = 0 ? ;
NO
S27
PERFORMING Read PROCESS ON Order (1)
S28
SHIFTING Order (2) AND SUBSEQUENT ORDERS BY 1
D o l=2, m Order (l−1)=Order (l) ;
m−1 → m ;

FIG. 19

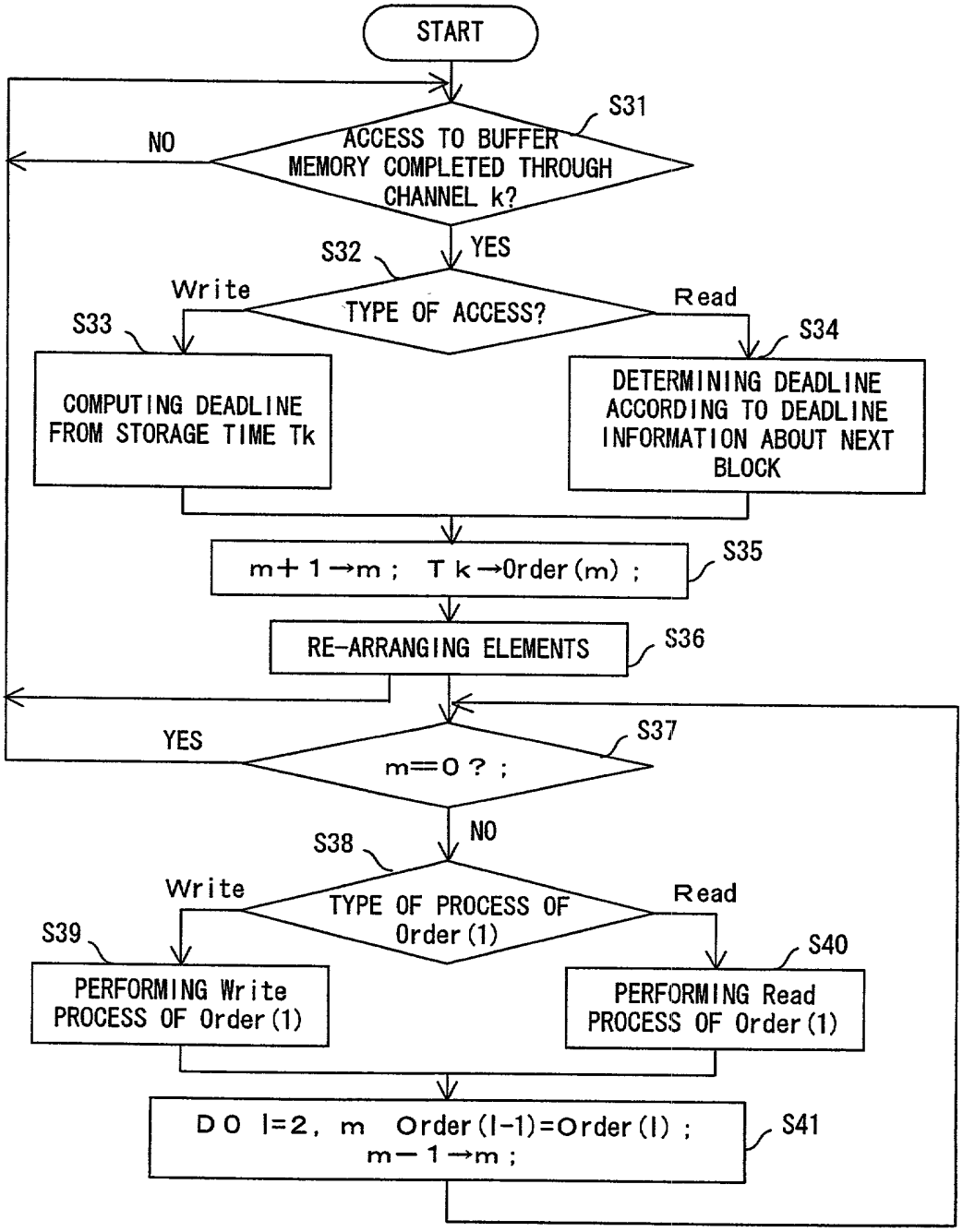

START
S31
ACCESS TO BUFFER MEMORY COMPLETED THROUGH CHANNEL k?
NO
YES
S32
TYPE OF ACCESS?
Write
Read
S33
COMPUTING DEADLINE FROM STORAGE TIME Tk
S34
DETERMINING DEADLINE ACCORDING TO DEADLINE INFORMATION ABOUT NEXT BLOCK
m+1→m ; Tk→Order(m) ;
S35
RE-ARRANGING ELEMENTS
S36
YES
m==0 ? ;
S37
NO
S38
TYPE OF PROCESS OF Order(1)
Write
Read
S39
PERFORMING Write PROCESS OF Order(1)
S40
PERFORMING Read PROCESS OF Order(1)
DO I=2, m Order(I-1)=Order(I) ;
m-1→m ;
S41

FIG. 20

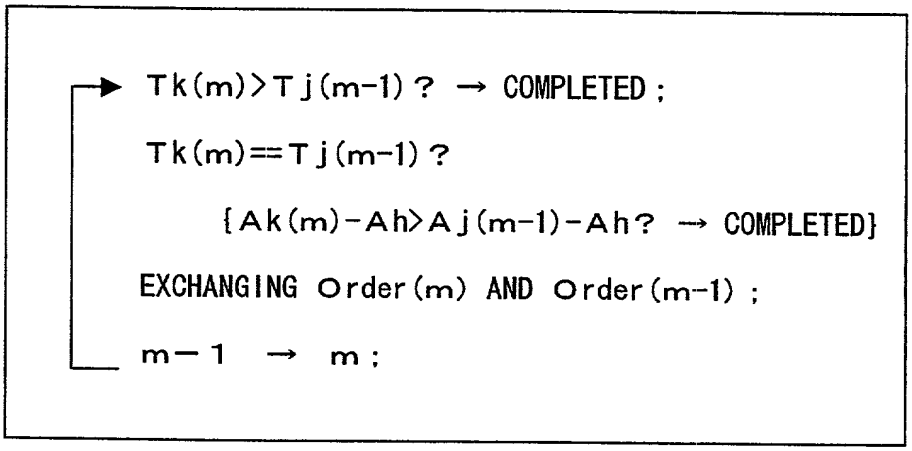

Tk(m)>Tj(m-1)? → COMPLETED;
Tk(m)==Tj(m-1)?
{Ak(m)-Ah>Aj(m-1)-Ah? → COMPLETED}
EXCHANGING Order(m) AND Order(m-1);
m-1 → m;

FIG. 21

Tk(m)>Tj(m／2)？ → >Tj(3m／4)？

CONTINUING

<Tj(3m／4)？

Tk(m)<Tj(m／2)？ → >Tj(m／4)？

CONTINUING

<Tj(m／4)？

INSERTING Order(m) IN ORDER OF DETERMINATION；

FIG. 22

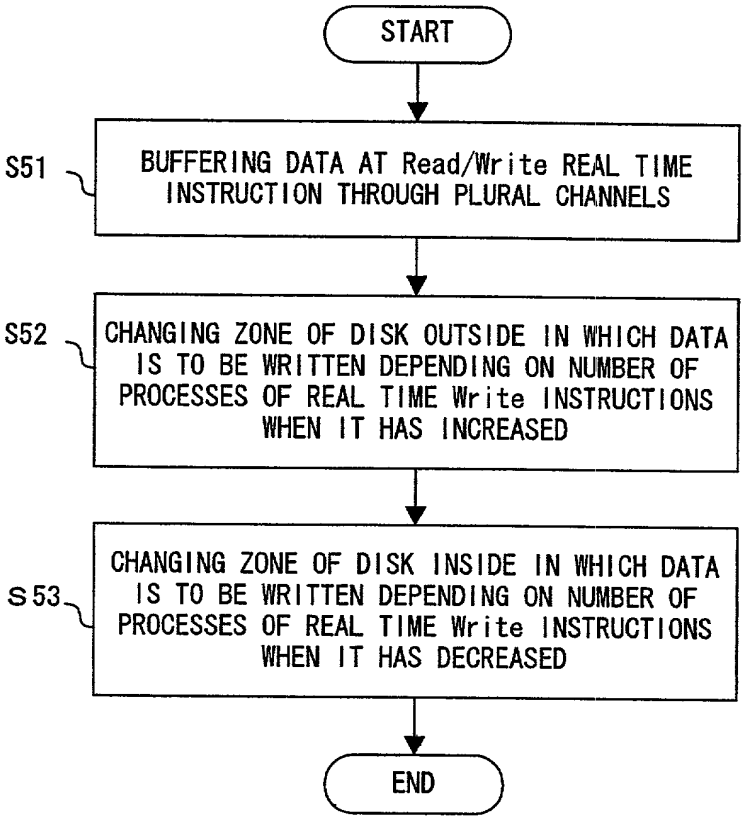

START
S51
BUFFERING DATA AT Read/Write REAL TIME INSTRUCTION THROUGH PLURAL CHANNELS
S52
CHANGING ZONE OF DISK OUTSIDE IN WHICH DATA IS TO BE WRITTEN DEPENDING ON NUMBER OF PROCESSES OF REAL TIME Write INSTRUCTIONS WHEN IT HAS INCREASED
S53
CHANGING ZONE OF DISK INSIDE IN WHICH DATA IS TO BE WRITTEN DEPENDING ON NUMBER OF PROCESSES OF REAL TIME Write INSTRUCTIONS WHEN IT HAS DECREASED
END

FIG. 23

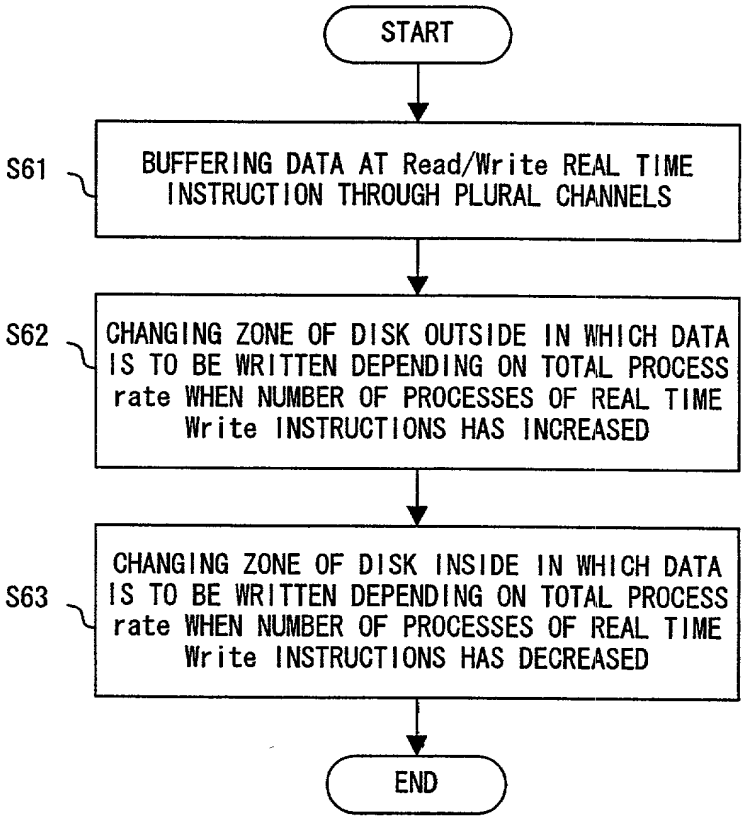

START
S61
BUFFERING DATA AT Read/Write REAL TIME INSTRUCTION THROUGH PLURAL CHANNELS
S62
CHANGING ZONE OF DISK OUTSIDE IN WHICH DATA IS TO BE WRITTEN DEPENDING ON TOTAL PROCESS rate WHEN NUMBER OF PROCESSES OF REAL TIME Write INSTRUCTIONS HAS INCREASED
S63
CHANGING ZONE OF DISK INSIDE IN WHICH DATA IS TO BE WRITTEN DEPENDING ON TOTAL PROCESS rate WHEN NUMBER OF PROCESSES OF REAL TIME Write INSTRUCTIONS HAS DECREASED
END

FIG. 24

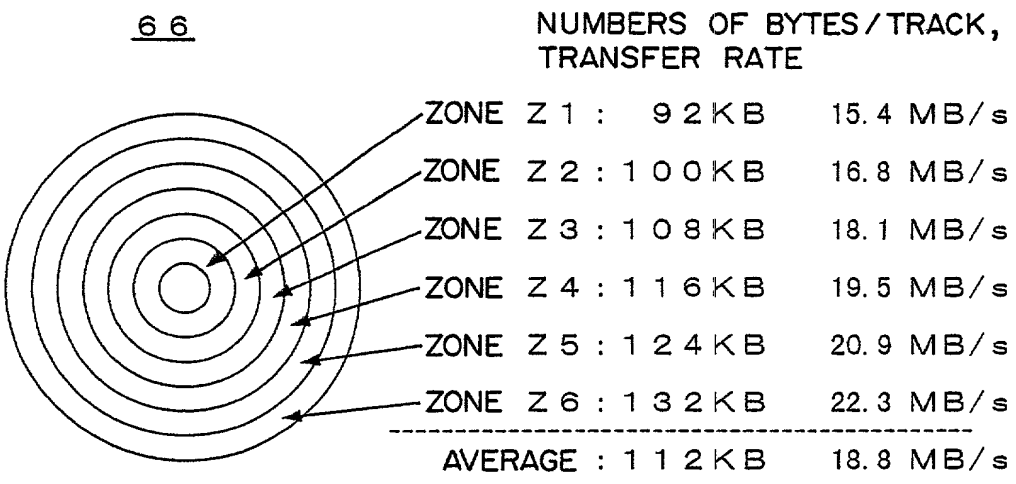

66
NUMBERS OF BYTES/TRACK,
TRANSFER RATE
ZONE Z1 : 92KB 15.4 MB/s
ZONE Z2 : 100KB 16.8 MB/s
ZONE Z3 : 108KB 18.1 MB/s
ZONE Z4 : 116KB 19.5 MB/s
ZONE Z5 : 124KB 20.9 MB/s
ZONE Z6 : 132KB 22.3 MB/s
AVERAGE : 112KB 18.8 MB/s

FIG. 25

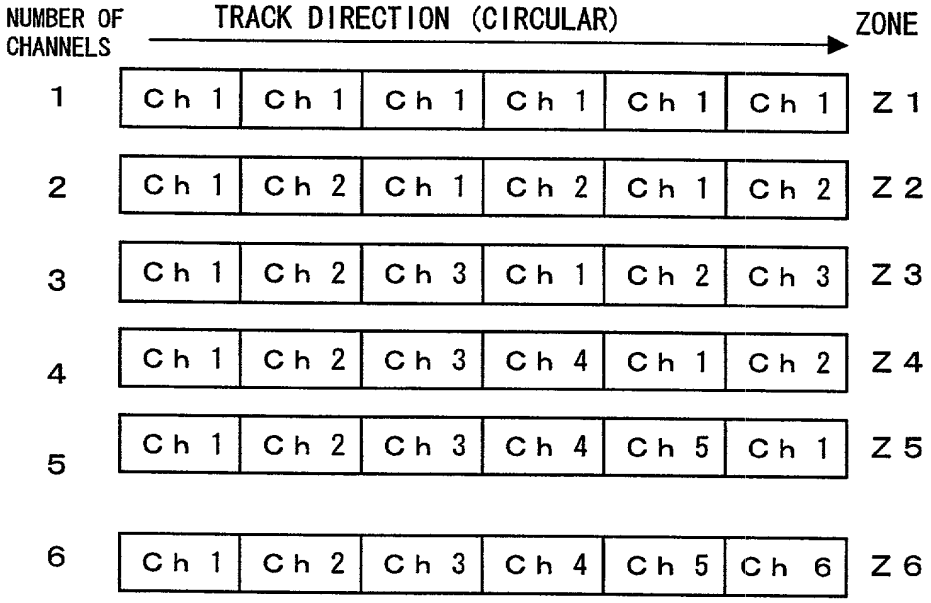

NUMBER OF CHANNELS
TRACK DIRECTION (CIRCULAR)
ZONE
1
Ch 1 Ch 1 Ch 1 Ch 1 Ch 1 Ch 1
Z 1
2
Ch 1 Ch 2 Ch 1 Ch 2 Ch 1 Ch 2
Z 2
3
Ch 1 Ch 2 Ch 3 Ch 1 Ch 2 Ch 3
Z 3
4
Ch 1 Ch 2 Ch 3 Ch 4 Ch 1 Ch 2
Z 4
5
Ch 1 Ch 2 Ch 3 Ch 4 Ch 5 Ch 1
Z 5
6
Ch 1 Ch 2 Ch 3 Ch 4 Ch 5 Ch 6
Z 6

FIG. 26

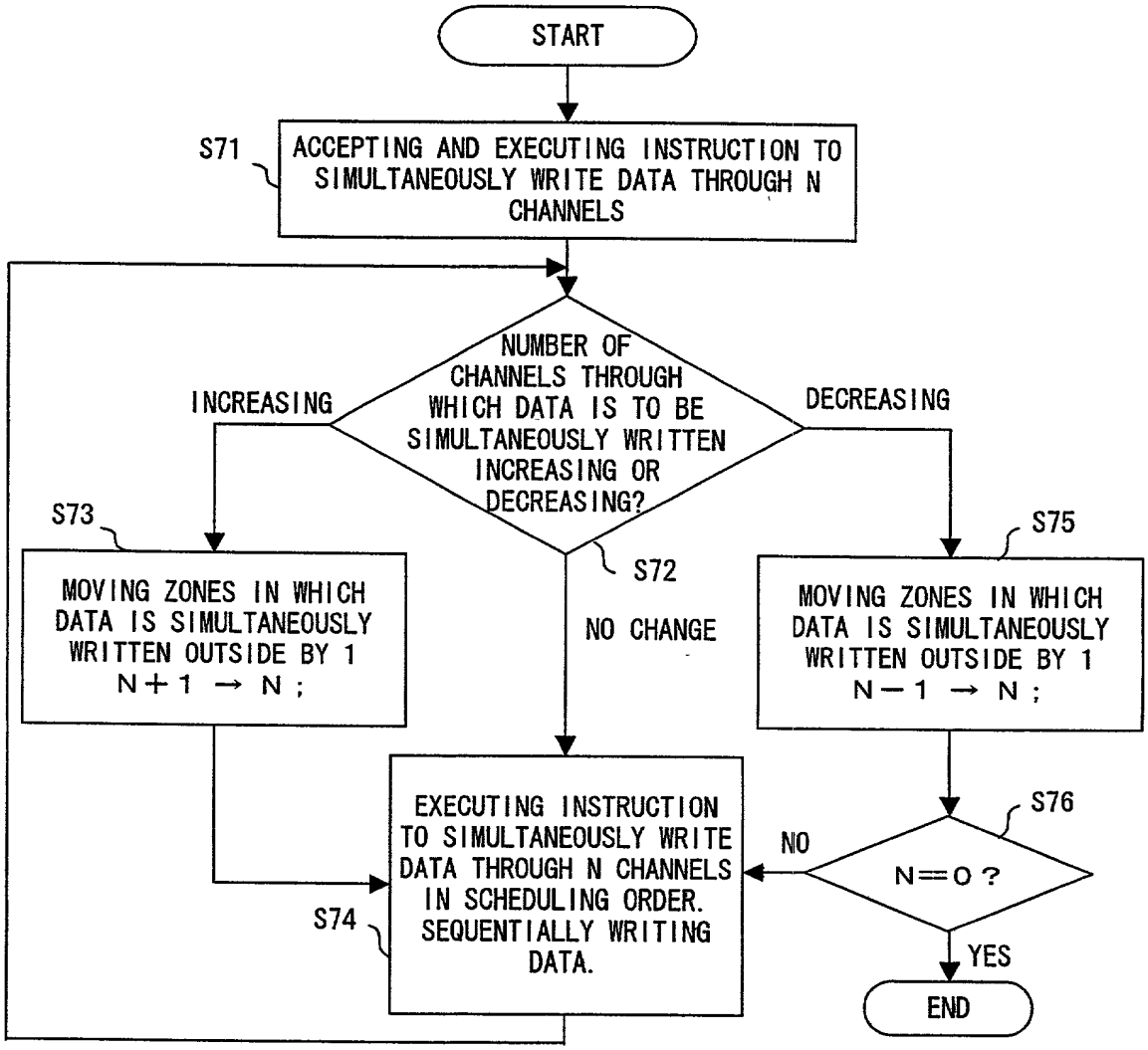

START
S71
ACCEPTING AND EXECUTING INSTRUCTION TO SIMULTANEOUSLY WRITE DATA THROUGH N CHANNELS
NUMBER OF CHANNELS THROUGH WHICH DATA IS TO BE SIMULTANEOUSLY WRITTEN INCREASING OR DECREASING?
S72
INCREASING
DECREASING
NO CHANGE
S73
MOVING ZONES IN WHICH DATA IS SIMULTANEOUSLY WRITTEN OUTSIDE BY 1
N+1 → N ;
S75
MOVING ZONES IN WHICH DATA IS SIMULTANEOUSLY WRITTEN OUTSIDE BY 1
N−1 → N ;
S76
N==0 ?
NO
YES
END
EXECUTING INSTRUCTION TO SIMULTANEOUSLY WRITE DATA THROUGH N CHANNELS IN SCHEDULING ORDER. SEQUENTIALLY WRITING DATA.
S74

FIG. 27

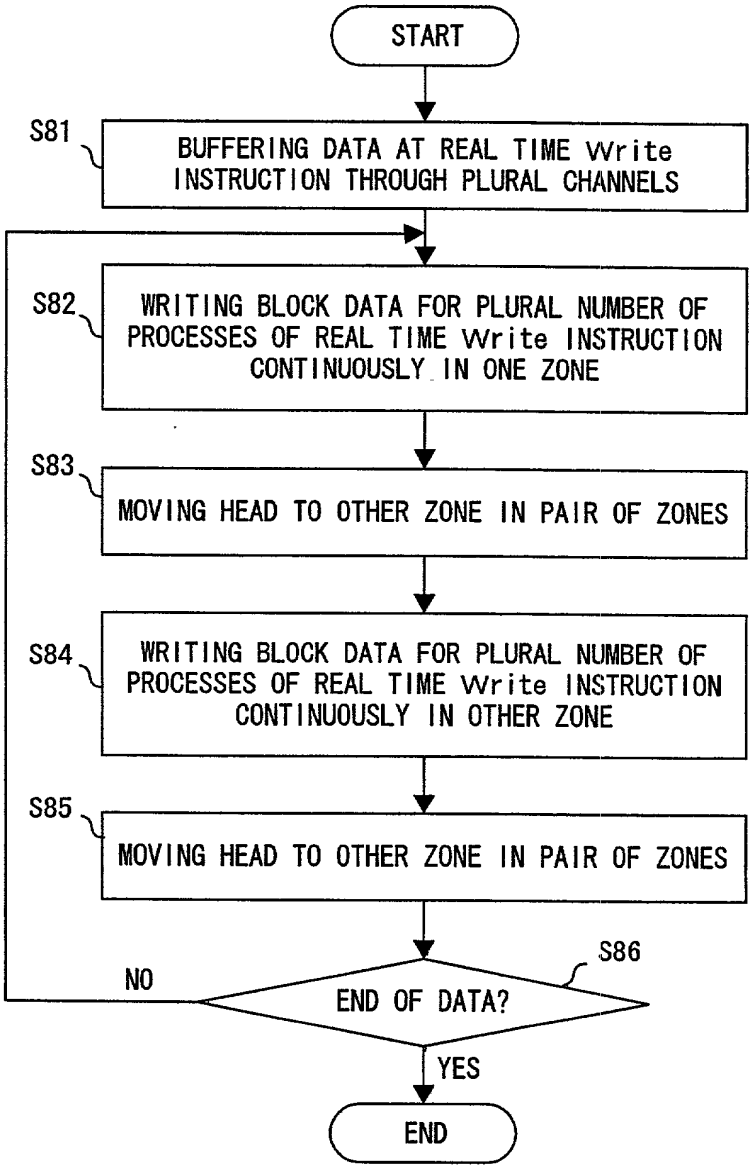

START
S81
BUFFERING DATA AT REAL TIME Write INSTRUCTION THROUGH PLURAL CHANNELS
S82
WRITING BLOCK DATA FOR PLURAL NUMBER OF PROCESSES OF REAL TIME Write INSTRUCTION CONTINUOUSLY IN ONE ZONE
S83
MOVING HEAD TO OTHER ZONE IN PAIR OF ZONES
S84
WRITING BLOCK DATA FOR PLURAL NUMBER OF PROCESSES OF REAL TIME Write INSTRUCTION CONTINUOUSLY IN OTHER ZONE
S85
MOVING HEAD TO OTHER ZONE IN PAIR OF ZONES
S86
NO
END OF DATA?
YES
END

FIG. 28

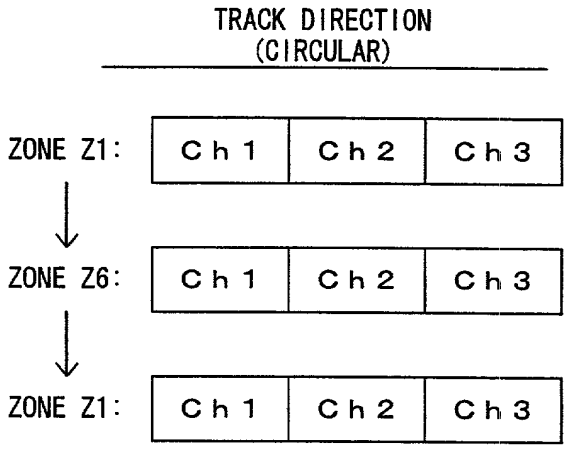

TRACK DIRECTION
(CIRCULAR)
ZONE Z1:
Ch 1
Ch 2
Ch 3
ZONE Z6:
Ch 1
Ch 2
Ch 3
ZONE Z1:
Ch 1
Ch 2
Ch 3

FIG. 29

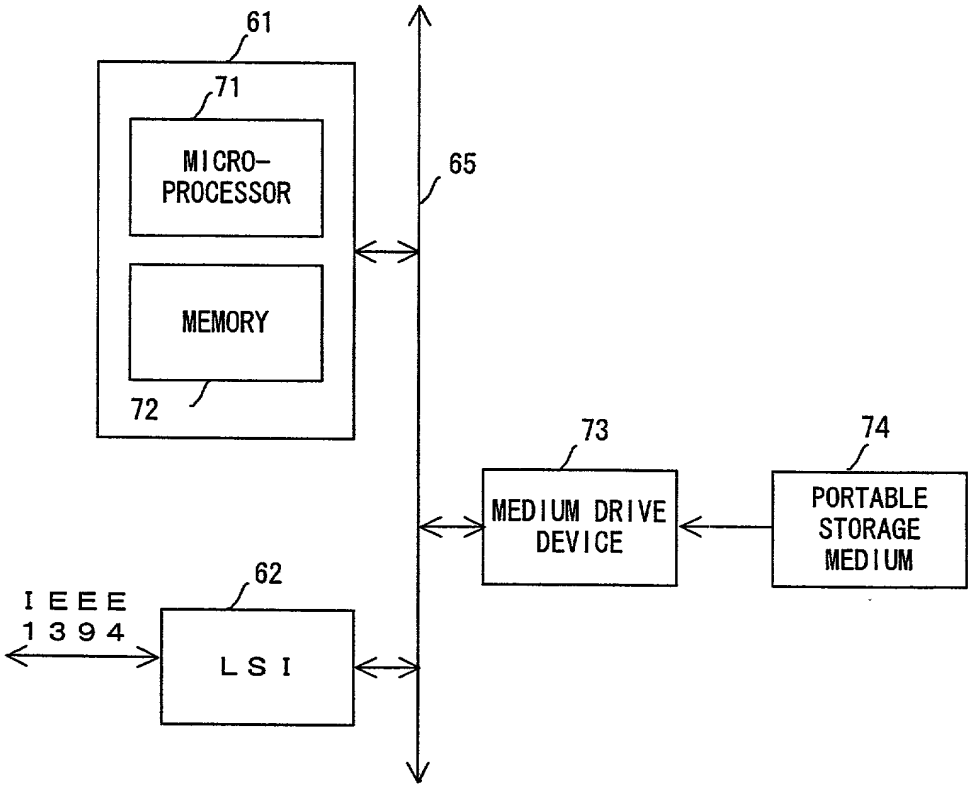

61
71
MICRO-
PROCESSOR
MEMORY
72
65
73
MEDIUM DRIVE
DEVICE
74
PORTABLE
STORAGE
MEDIUM
62
IEEE
1394
LSI

FIG. 30

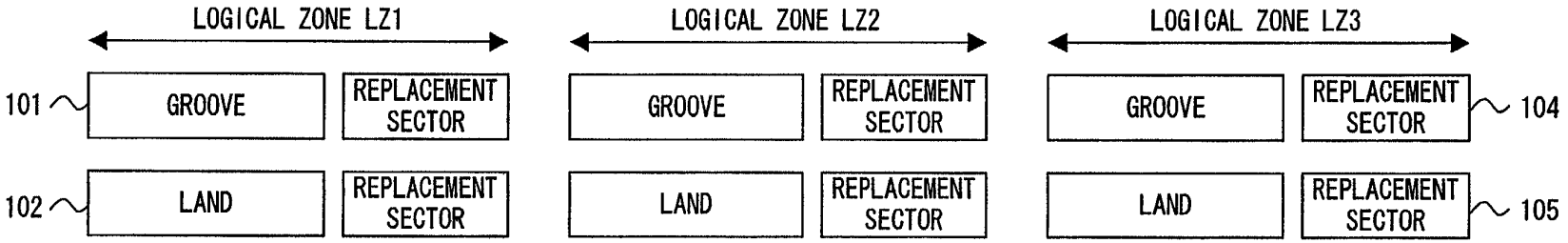

LOGICAL ZONE LZ1
LOGICAL ZONE LZ2
LOGICAL ZONE LZ3
101
GROOVE
REPLACEMENT SECTOR
GROOVE
REPLACEMENT SECTOR
GROOVE
REPLACEMENT SECTOR
104
102
LAND
REPLACEMENT SECTOR
LAND
REPLACEMENT SECTOR
LAND
REPLACEMENT SECTOR
105

FIG. 31

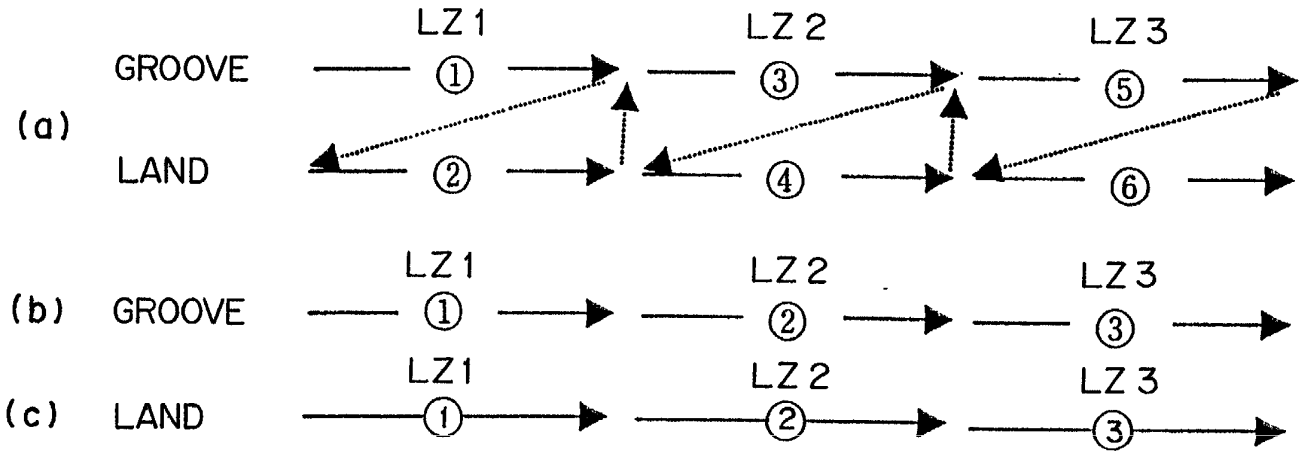


FIG. 32

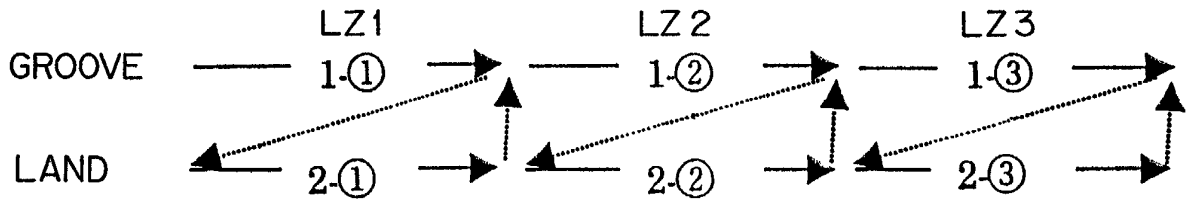


FIG. 33

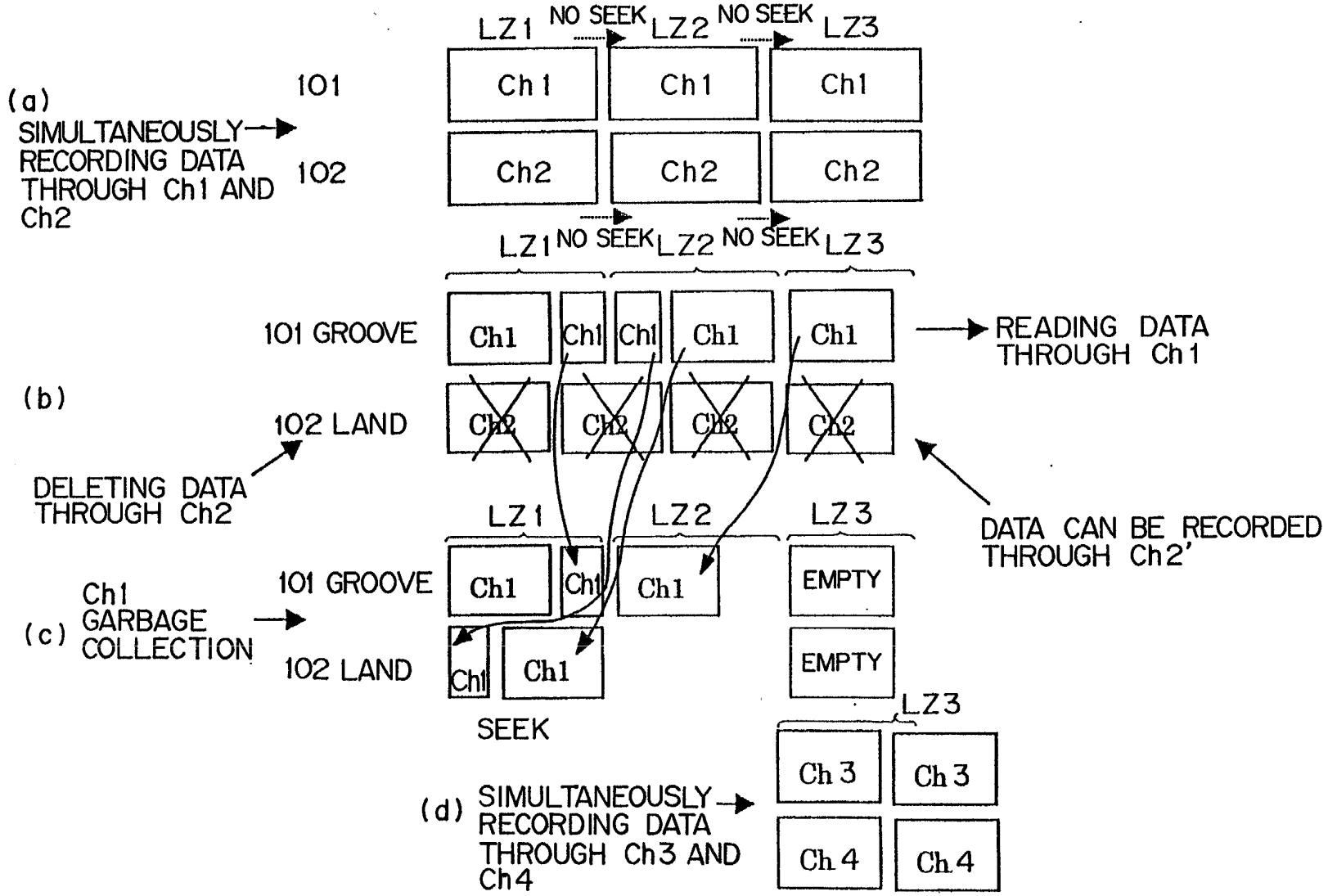

LZ1
NO SEEK
LZ2
NO SEEK
LZ3
(a)
101
SIMULTANEOUSLY RECORDING DATA THROUGH Ch1 AND Ch2
102
Ch1
Ch2
(b)
101 GROOVE
102 LAND
DELETING DATA THROUGH Ch2
READING DATA THROUGH Ch1
DATA CAN BE RECORDED THROUGH Ch2'
(c)
Ch1 GARBAGE COLLECTION
EMPTY
SEEK
(d)
SIMULTANEOUSLY RECORDING DATA THROUGH Ch3 AND Ch4
Ch3
Ch4

FIG. 34

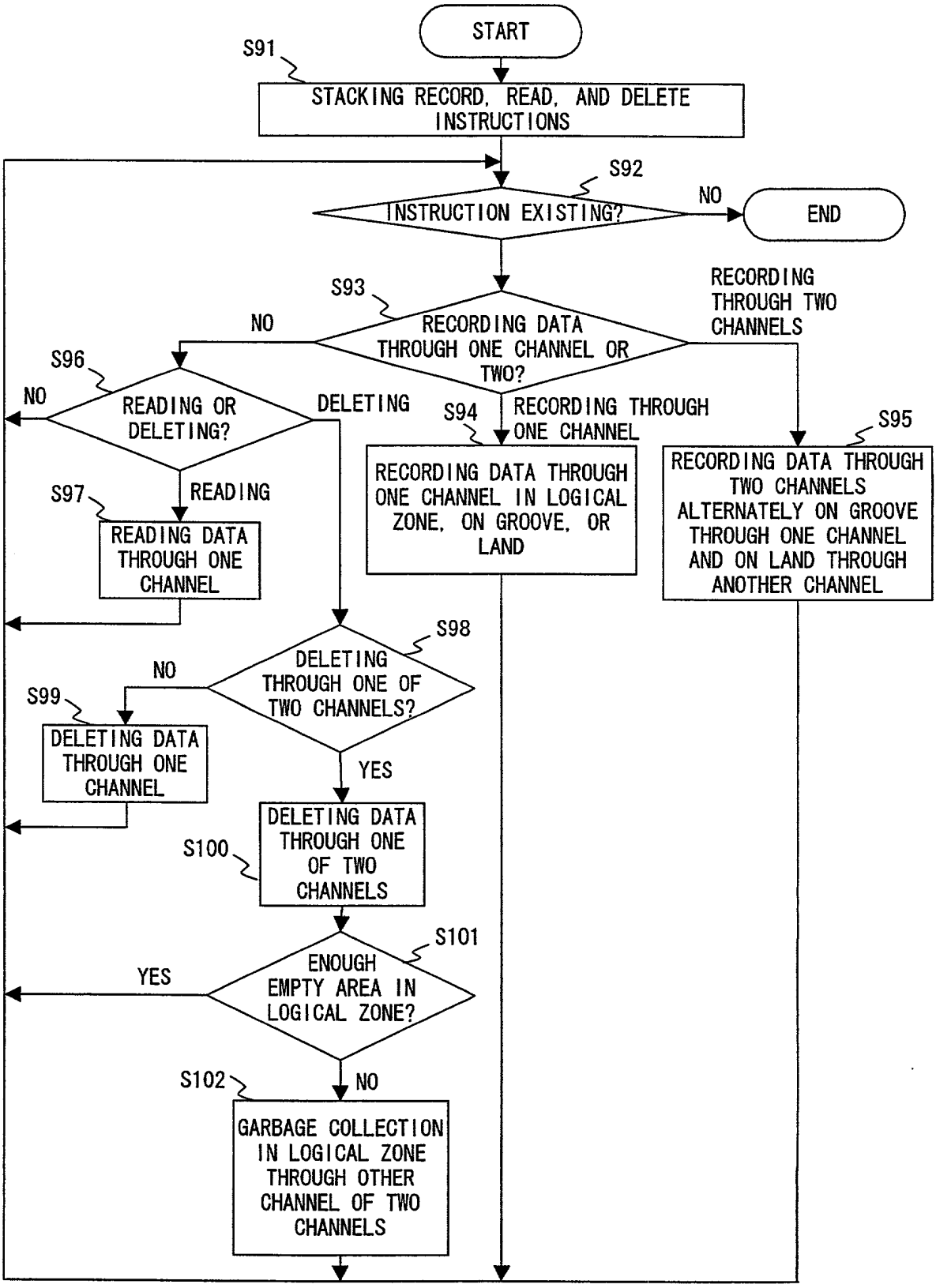

START
S91
STACKING RECORD, READ, AND DELETE INSTRUCTIONS
S92
INSTRUCTION EXISTING?
NO
END
S93
RECORDING DATA THROUGH ONE CHANNEL OR TWO?
NO
RECORDING THROUGH TWO CHANNELS
S96
READING OR DELETING?
NO
DELETING
S94
RECORDING THROUGH ONE CHANNEL
S95
RECORDING DATA THROUGH ONE CHANNEL IN LOGICAL ZONE, ON GROOVE, OR LAND
RECORDING DATA THROUGH TWO CHANNELS ALTERNATELY ON GROOVE THROUGH ONE CHANNEL AND ON LAND THROUGH ANOTHER CHANNEL
S97
READING
READING DATA THROUGH ONE CHANNEL
S98
DELETING THROUGH ONE OF TWO CHANNELS?
NO
S99
DELETING DATA THROUGH ONE CHANNEL
YES
DELETING DATA THROUGH ONE OF TWO CHANNELS
S100
S101
ENOUGH EMPTY AREA IN LOGICAL ZONE?
YES
S102
NO
GARBAGE COLLECTION IN LOGICAL ZONE THROUGH OTHER CHANNEL OF TWO CHANNELS

FIG. 35

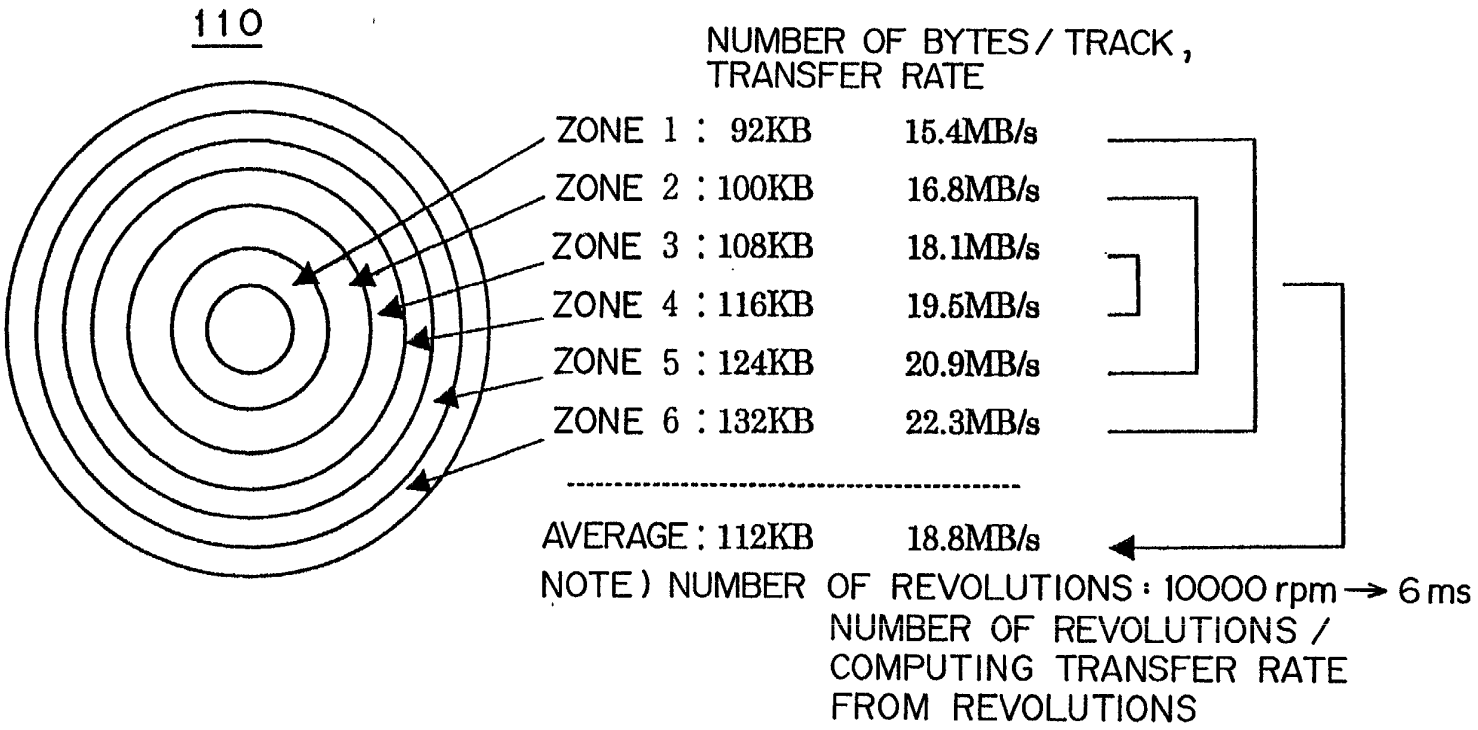

110
NUMBER OF BYTES / TRACK,
TRANSFER RATE
ZONE 1 : 92KB 15.4MB/s
ZONE 2 : 100KB 16.8MB/s
ZONE 3 : 108KB 18.1MB/s
ZONE 4 : 116KB 19.5MB/s
ZONE 5 : 124KB 20.9MB/s
ZONE 6 : 132KB 22.3MB/s
AVERAGE : 112KB 18.8MB/s
NOTE) NUMBER OF REVOLUTIONS : 10000 rpm → 6 ms
NUMBER OF REVOLUTIONS /
COMPUTING TRANSFER RATE
FROM REVOLUTIONS

FIG. 36

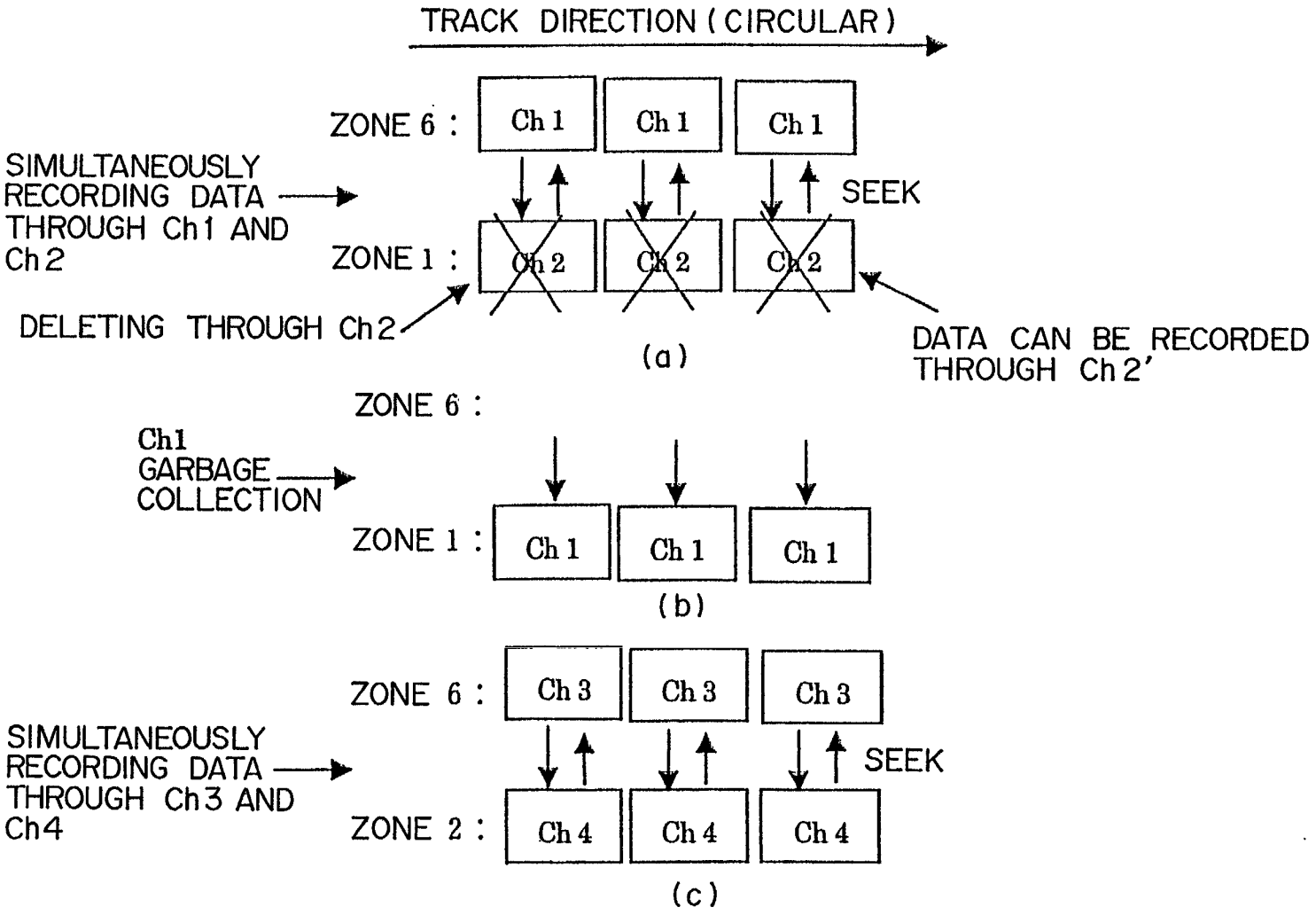

TRACK DIRECTION (CIRCULAR)
ZONE 6 :
Ch 1
Ch 1
Ch 1
SIMULTANEOUSLY RECORDING DATA THROUGH Ch1 AND Ch2
SEEK
ZONE 1 :
Ch 2
Ch 2
Ch 2
DELETING THROUGH Ch2
DATA CAN BE RECORDED THROUGH Ch2'
(a)
ZONE 6 :
Ch1 GARBAGE COLLECTION
ZONE 1 :
Ch 1
Ch 1
Ch 1
(b)
ZONE 6 :
Ch 3
Ch 3
Ch 3
SIMULTANEOUSLY RECORDING DATA THROUGH Ch3 AND Ch4
SEEK
ZONE 2 :
Ch 4
Ch 4
Ch 4
(c)

FIG. 37

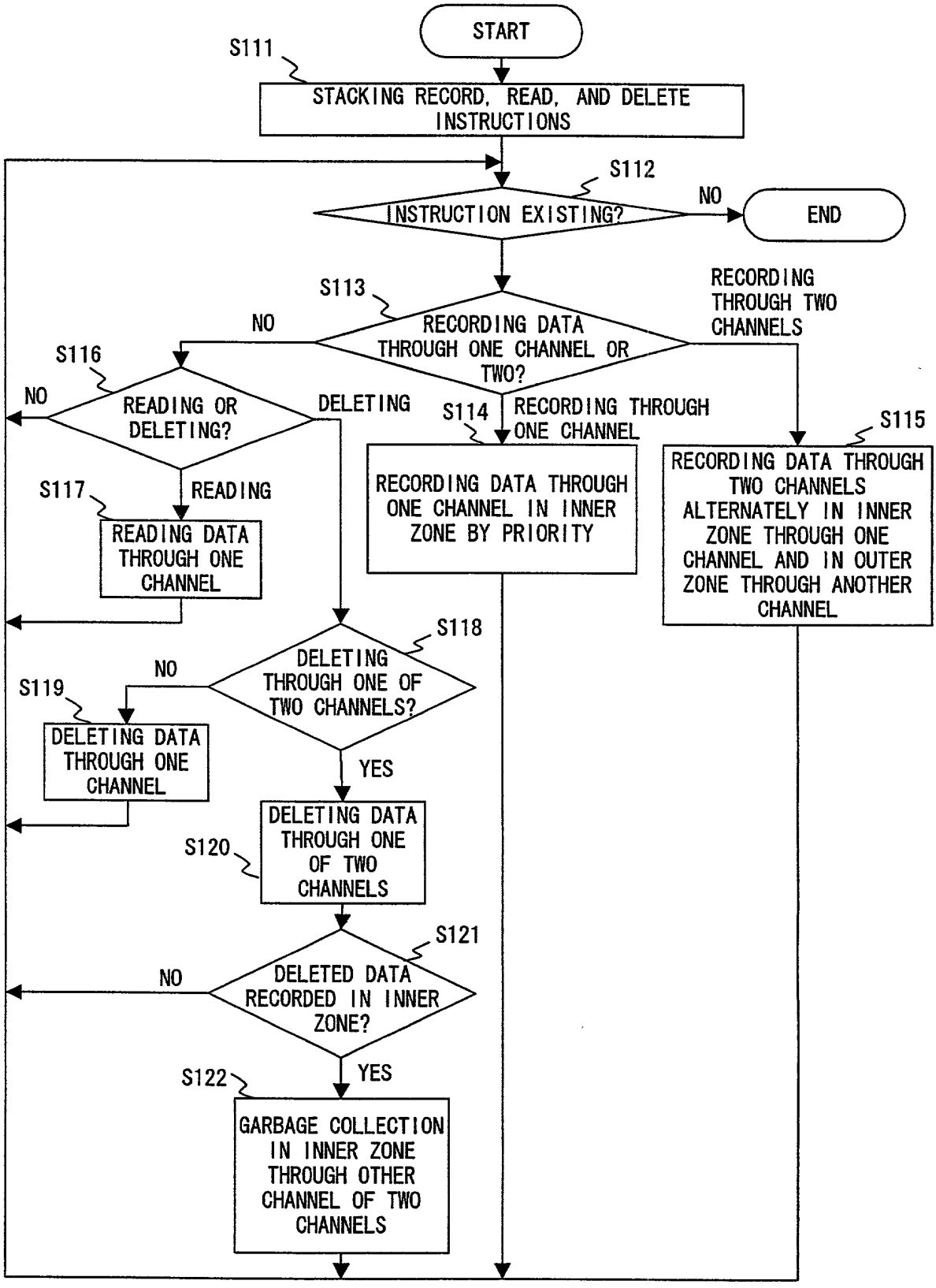

START
S111
STACKING RECORD, READ, AND DELETE INSTRUCTIONS
S112
INSTRUCTION EXISTING?
NO
END
S113
RECORDING DATA THROUGH ONE CHANNEL OR TWO?
NO
RECORDING THROUGH TWO CHANNELS
S116
READING OR DELETING?
NO
DELETING
S114
RECORDING THROUGH ONE CHANNEL
S115
S117
READING
READING DATA THROUGH ONE CHANNEL
RECORDING DATA THROUGH ONE CHANNEL IN INNER ZONE BY PRIORITY
RECORDING DATA THROUGH TWO CHANNELS ALTERNATELY IN INNER ZONE THROUGH ONE CHANNEL AND IN OUTER ZONE THROUGH ANOTHER CHANNEL
S118
DELETING THROUGH ONE OF TWO CHANNELS?
NO
S119
DELETING DATA THROUGH ONE CHANNEL
YES
S120
DELETING DATA THROUGH ONE OF TWO CHANNELS
S121
DELETED DATA RECORDED IN INNER ZONE?
NO
YES
S122
GARBAGE COLLECTION IN INNER ZONE THROUGH OTHER CHANNEL OF TWO CHANNELS

FIG. 38

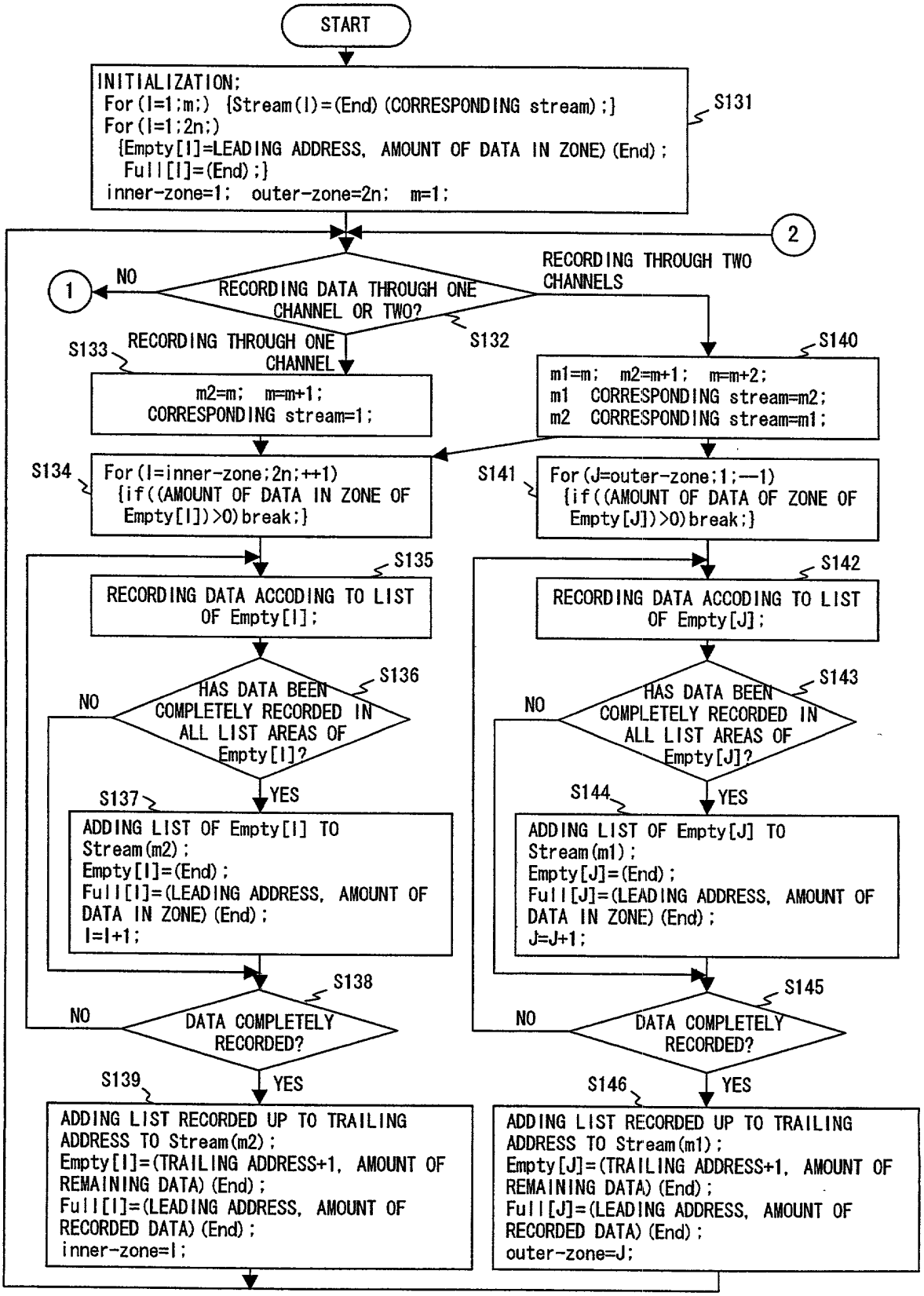

START
INITIALIZATION;
For(I=1;m;) {Stream(I)=(End) (CORRESPONDING stream);}
For(I=1;2n;)
{Empty[I]=LEADING ADDRESS, AMOUNT OF DATA IN ZONE) (End);
Full[I]=(End);}
inner-zone=1; outer-zone=2n; m=1;
S131
2
1
NO
RECORDING DATA THROUGH ONE CHANNEL OR TWO?
S132
RECORDING THROUGH TWO CHANNELS
S133
RECORDING THROUGH ONE CHANNEL
m2=m; m=m+1;
CORRESPONDING stream=1;
S140
m1=m; m2=m+1; m=m+2;
m1 CORRESPONDING stream=m2;
m2 CORRESPONDING stream=m1;
S134
For(I=inner-zone;2n;++1)
{if((AMOUNT OF DATA IN ZONE OF Empty[I])>0)break;}
S141
For(J=outer-zone;1;—1)
{if((AMOUNT OF DATA OF ZONE OF Empty[J])>0)break;}
S135
RECORDING DATA ACCODING TO LIST OF Empty[I];
S142
RECORDING DATA ACCODING TO LIST OF Empty[J];
S136
NO
HAS DATA BEEN COMPLETELY RECORDED IN ALL LIST AREAS OF Empty[I]?
YES
S143
NO
HAS DATA BEEN COMPLETELY RECORDED IN ALL LIST AREAS OF Empty[J]?
YES
S137
ADDING LIST OF Empty[I] TO Stream(m2);
Empty[I]=(End);
Full[I]=(LEADING ADDRESS, AMOUNT OF DATA IN ZONE) (End);
I=I+1;
S144
ADDING LIST OF Empty[J] TO Stream(m1);
Empty[J]=(End);
Full[J]=(LEADING ADDRESS, AMOUNT OF DATA IN ZONE) (End);
J=J+1;
S138
NO
DATA COMPLETELY RECORDED?
YES
S145
NO
DATA COMPLETELY RECORDED?
YES
S139
ADDING LIST RECORDED UP TO TRAILING ADDRESS TO Stream(m2);
Empty[I]=(TRAILING ADDRESS+1, AMOUNT OF REMAINING DATA) (End);
Full[I]=(LEADING ADDRESS, AMOUNT OF RECORDED DATA) (End);
inner-zone=I;
S146
ADDING LIST RECORDED UP TO TRAILING ADDRESS TO Stream(m1);
Empty[J]=(TRAILING ADDRESS+1, AMOUNT OF REMAINING DATA) (End);
Full[J]=(LEADING ADDRESS, AMOUNT OF RECORDED DATA) (End);
outer-zone=J;

FIG. 39

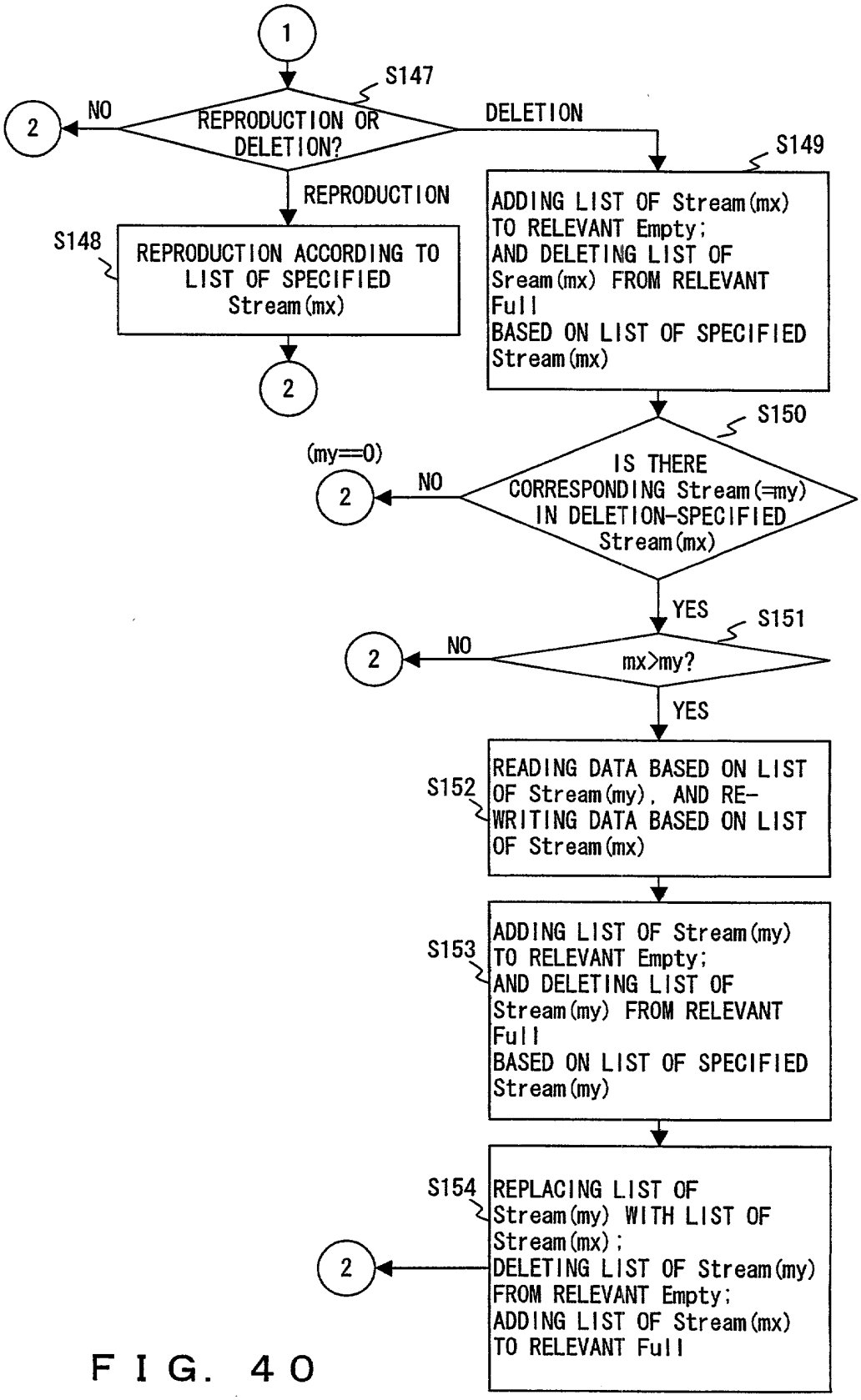

1
S147
2
NO
REPRODUCTION OR DELETION?
DELETION
S149
REPRODUCTION
S148
REPRODUCTION ACCORDING TO LIST OF SPECIFIED Stream(mx)
ADDING LIST OF Stream(mx) TO RELEVANT Empty; AND DELETING LIST OF Sream(mx) FROM RELEVANT Full BASED ON LIST OF SPECIFIED Stream(mx)
2
S150
(my==0)
2
NO
IS THERE CORRESPONDING Stream(=my) IN DELETION-SPECIFIED Stream(mx)
YES
S151
2
NO
mx>my?
YES
S152
READING DATA BASED ON LIST OF Stream(my), AND RE-WRITING DATA BASED ON LIST OF Stream(mx)
S153
ADDING LIST OF Stream(my) TO RELEVANT Empty; AND DELETING LIST OF Stream(my) FROM RELEVANT Full BASED ON LIST OF SPECIFIED Stream(my)
S154
2
REPLACING LIST OF Stream(my) WITH LIST OF Stream(mx); DELETING LIST OF Stream(my) FROM RELEVANT Empty; ADDING LIST OF Stream(mx) TO RELEVANT Full

FIG. 40

120

STREAM LIST	(LEADING ADDRESS, AMOUNT OF DATA) → COMPLETION, CORRESPONDING stream FOR RECORDING DATA THROUGH TWO CHANNELS
Stream(1)	(Add, Data) → (Add, Data) →···→ End.m?
⋮	⋮
Stream(m)	(Add, Data) → (Add, Data) →···→ End.m?

FIG. 41

130

INNER/ OUTER	ZONE	EMPTY/FULL LIST	(LEADING ADDRESS, AMOUNT OF DATA) → COMPLETION : LIST STRUCTURE
INNER ZONE	1	Empty[1]	(Add, Data) → (Add, Data) →···→ End
		Full[1]	(Add, Data) → (Add, Data) →···→ End
	⋮	⋮	⋮
	n	Empty[n]	(Add, Data) → End (INITIAL VALUE)
		Full[n]	End (INITIAL VALUE)
OUTER ZONE	n+1	Empty[n+1]	(Add, Data) → End (INITIAL VALUE)
		Full[n+1]	End (INITIAL VALUE)
	⋮	⋮	⋮
	2n	Empty[2n]	(Add, Data) → (Add, Data) →···→ End
		Full[2n]	(Add, Data) → (Add, Data) →···→ End

FIG. 42

ZONE no.	NUMBER OF BYTES/TRACK	NUMBER OF TRACKS	NUMBER OF SECTORS	SECTOR ADDRESS
1	92KB	1000	184k	1-184000
2	100KB	1000	200k	184000-384000
3	108KB	1000	216k	384001-600000
4	116KB	1000	232k	600001-832000
5	124KB	1000	248k	832001-1080000
6	132KB	1000	264k	1080001-1344000

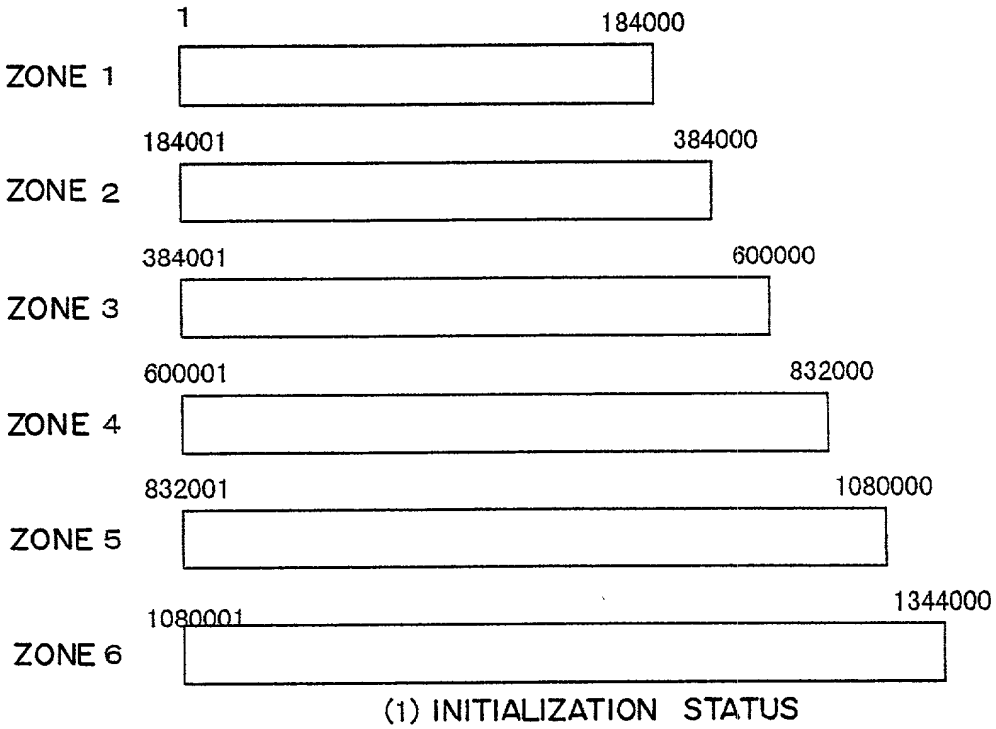
*: 512B/SECTOR

FIG. 43

150

	STORAGE ADDRESS	MEANING OF INFORMATION	STORED DATA (LEADING STORAGE DEVICE, AMOUNT OF DATA, NEXT STORAGE ADDRESS)
INITIALIZED AREA (120, 151, 130)	0	End	(0, 0, 0)
	1	Stream(1)	① (0, 0, 0) ③→ (1080001, 264000, 19) ⑤→ (1, 184000, 20)
	2	Stream(2)	① (0, 0, 0) ②→ (1, 184000, 18) ④→ (0, 0, 0)
	3	Stream(3)	① (0, 0, 0) ⑦→ (1080001, 256000, 22)
	4	Stream(4)	① (0, 0, 0) ⑥→ (300001, 84000, 21)
	5	Stream(5)	① (0, 0, 0)
	6	Empty[1]	① (1, 184000, 0) ②→ (0, 0, 0) ④→ (1, 184000, 0) ⑤→ (0, 0, 0)
	7	Empty[2]	① (184001, 200000, 0) ③→ (300001, 84000. 0) ④→ (184001, 200000, 0) ⑤→ (300001, 84000, 0) ⑥→ (0, 0, 0)
	8	Empty[3]	① (384001, 216000, 0) ⑦→ (0, 0, 0)
	9	Empty[4]	① (600001, 232000, 0)
	10	Empty[5]	① (832001, 248000, 0 ③→ (876001, 204000, 0) ⑤→ (832001, 248000. 0) ⑦→ (876001, 204000, 0)
	11	Empty[6]	① (1080001, 256000, 0) ③→ (0, 0, 0) ⑤→ (1080001, 256000, 0) ⑦→ (0, 0, 0)
	12	Full[1]	① (1, 0, 0) ②→ (1, 184000, 0) ④→ (1, 0, 0) ⑤→ (1, 184000, 0)
	13	Full[2]	① (184001, 0, 0) ③→ (184001, 116000, 0) ④→ (184001, 0, 0) ⑤→ (184001, 11600, 0) ⑥→ (184001, 200000, 0)
	14	Full[3]	① (384001, 0, 0) ⑦→ (384001, 216000, 0)
	15	Full[4]	① (600001, 0, 0)
	16	Full[5]	① (832001, 0, 0) ③→ (832001, 44000, 0) ⑤→ (832001, 0, 0) ⑦→ (832001, 44000, 0)
	17	Full[6]	① (1080001, 0, 0) ③→ (1080001, 256000, 0) ⑥→ (1080001, 84001, 0) ⑦→ (1080001, 256000, 0)
EXTENSION AREA (152)	18	$Z_1 \rightarrow Z_2$	(184001, 0, 0) ③→ (184001, 116000, 0)
	19	$Z_6 \rightarrow Z_5$	(832001, 0, 0) ③→ (832001, 44000, 0)
	20	$Z_1 \rightarrow Z_2$	(184001, 0, 0) ⑤→ (184001, 116000, 0)
	21	$Z_2 \rightarrow Z_3$	(384001, 0, 0) ⑦→ (384001, 216000, 0)
	22	$Z_6 \rightarrow Z_5$	(832001, 0, 0) ⑦→ (832001, 44000, 0)
	23		
	24		
	25		
	26		
	27		

FIG. 44



(1) INITIALIZATION STATUS

FIG. 45

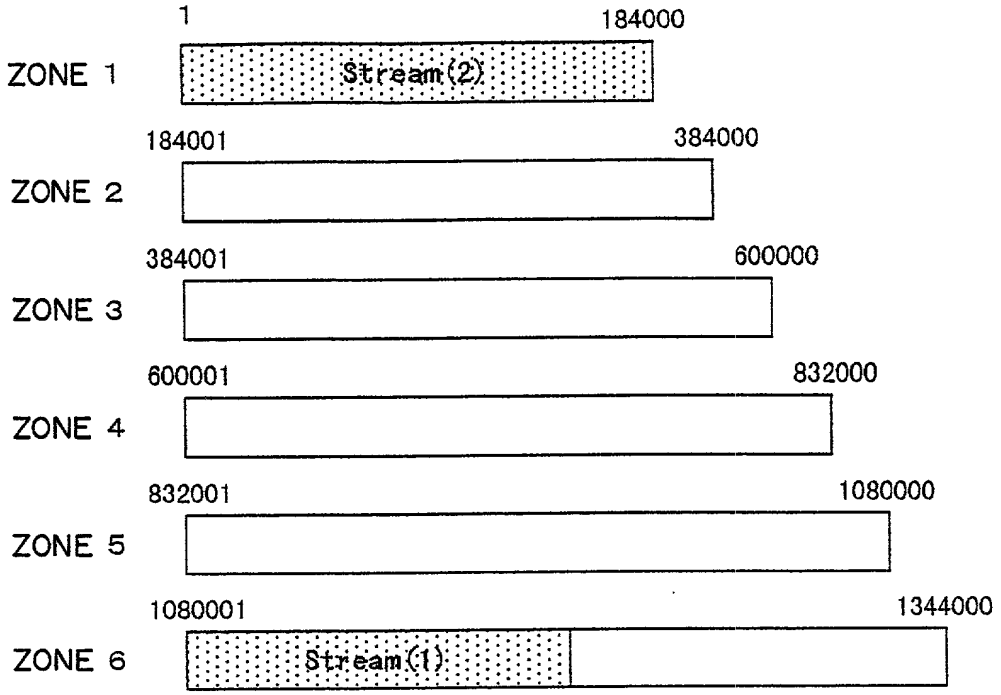

1
184000
ZONE 1
Stream(2)
184001
384000
ZONE 2
384001
600000
ZONE 3
600001
832000
ZONE 4
832001
1080000
ZONE 5
1080001
1344000
ZONE 6
Stream(1)

FIG. 46

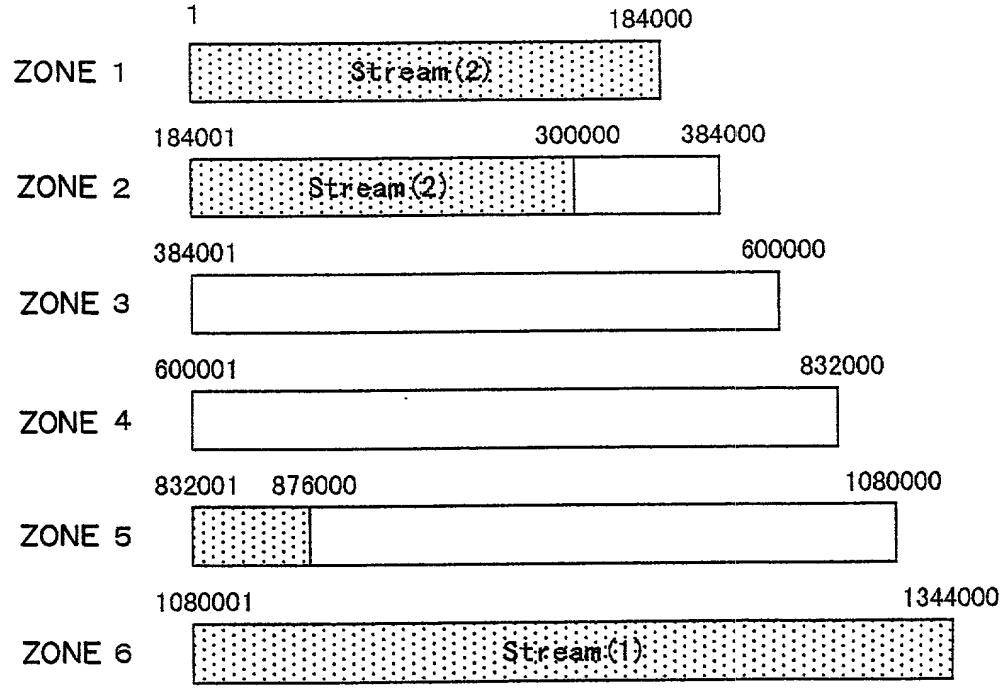

1
184000
ZONE 1
Stream(2)
184001
300000
384000
ZONE 2
Stream(2)
384001
600000
ZONE 3
600001
832000
ZONE 4
832001
876000
1080000
ZONE 5
1080001
1344000
ZONE 6
Stream(1)

FIG. 47

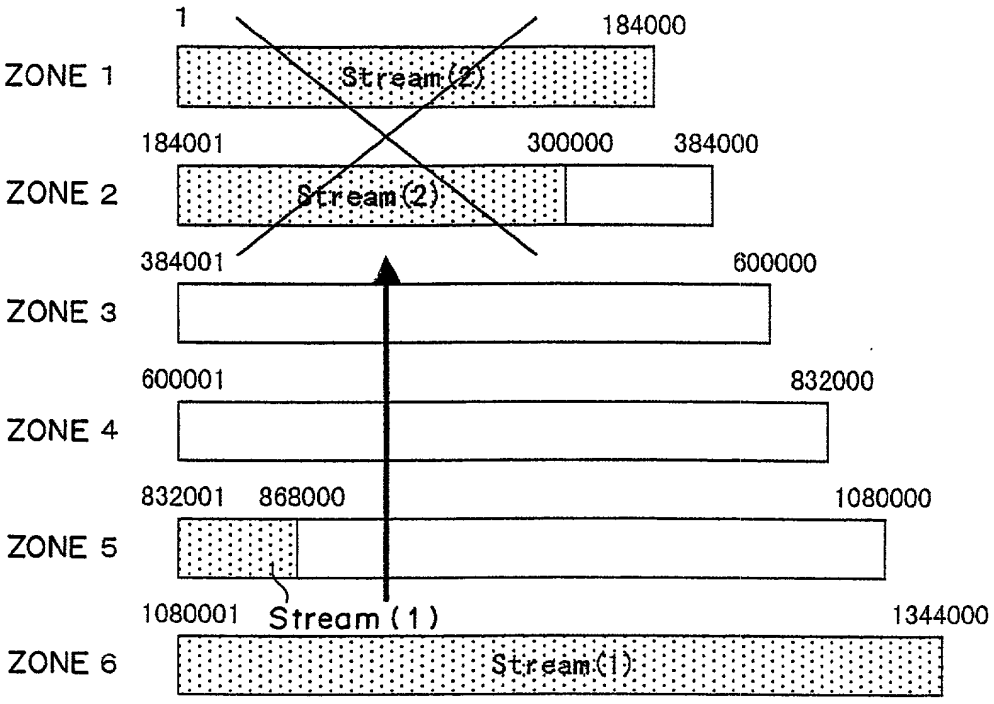

1
184000
ZONE 1
Stream(2)
184001
300000
384000
ZONE 2
Stream(2)
384001
600000
ZONE 3
600001
832000
ZONE 4
832001
868000
1080000
ZONE 5
1080001
Stream(1)
1344000
ZONE 6
Stream(1)

FIG. 48

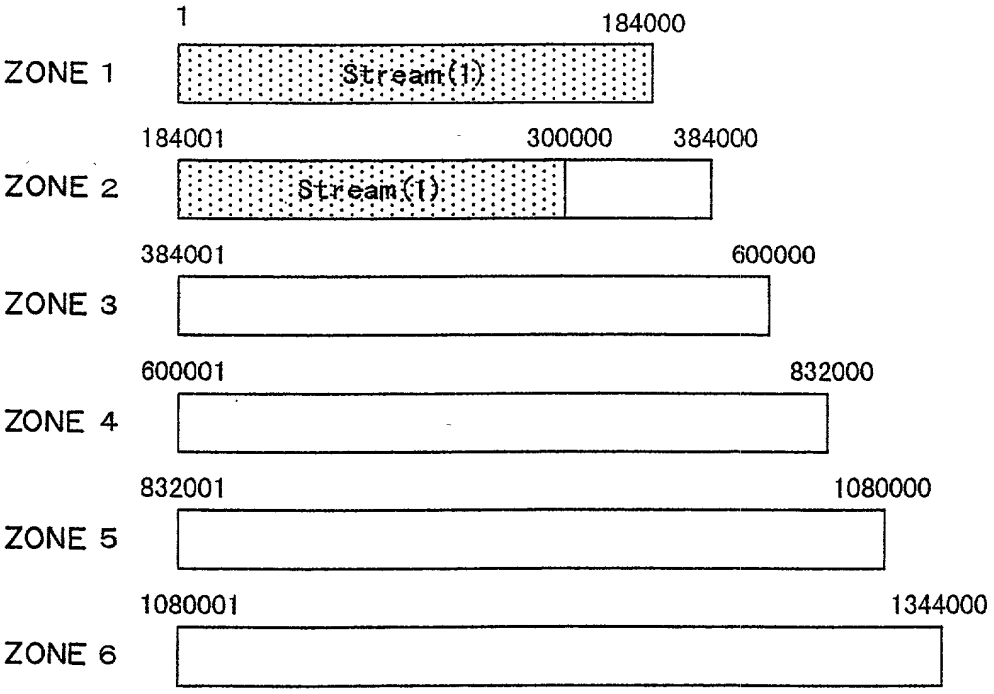

1
184000
ZONE 1
Stream(1)
184001
300000
384000
ZONE 2
Stream(1)
384001
600000
ZONE 3
600001
832000
ZONE 4
832001
1080000
ZONE 5
1080001
1344000
ZONE 6

FIG. 49

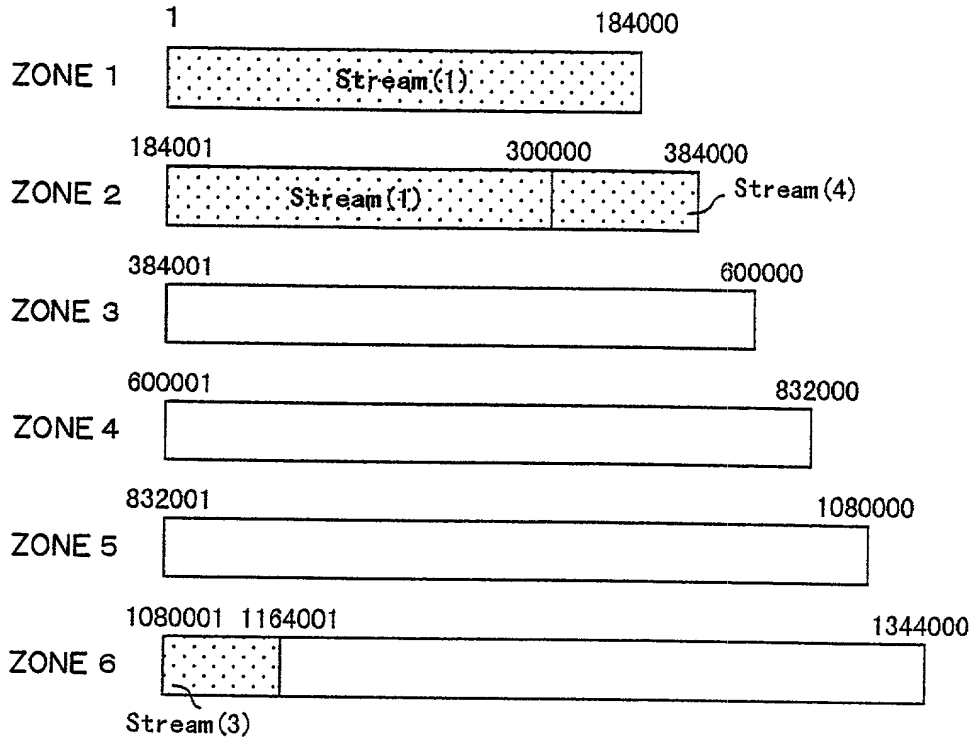

1
184000
ZONE 1
Stream(1)
184001
300000
384000
ZONE 2
Stream(1)
Stream(4)
384001
600000
ZONE 3
600001
832000
ZONE 4
832001
1080000
ZONE 5
1080001
1164001
1344000
ZONE 6
Stream(3)

FIG. 50

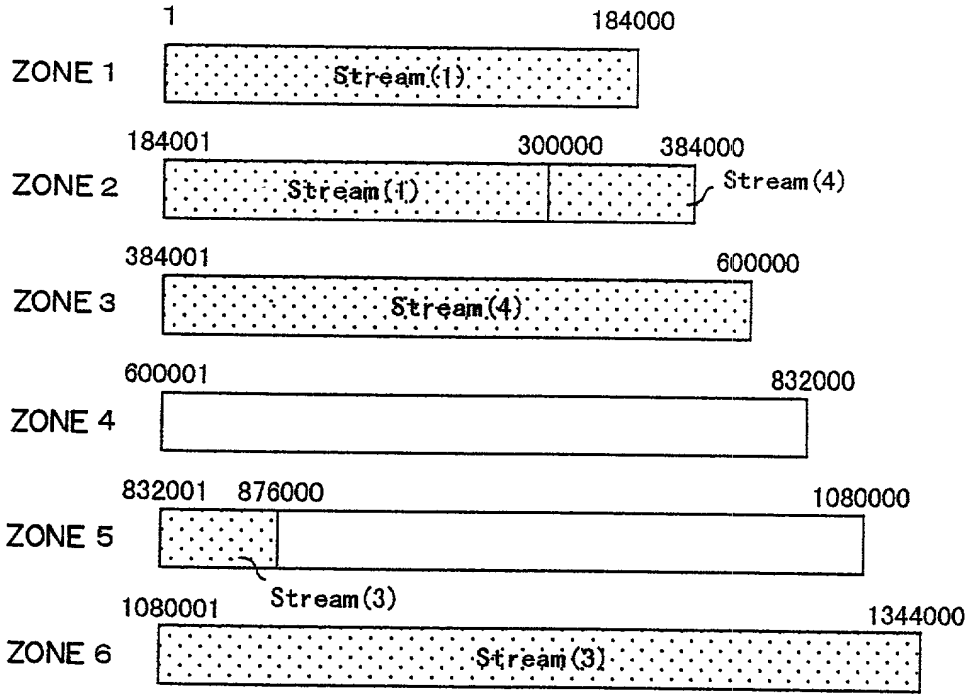

1
184000
ZONE 1
Stream(1)
184001
300000
384000
ZONE 2
Stream(1)
Stream(4)
384001
600000
ZONE 3
Stream(4)
600001
832000
ZONE 4
832001
876000
1080000
ZONE 5
Stream(3)
1080001
1344000
ZONE 6
Stream(3)

FIG. 51

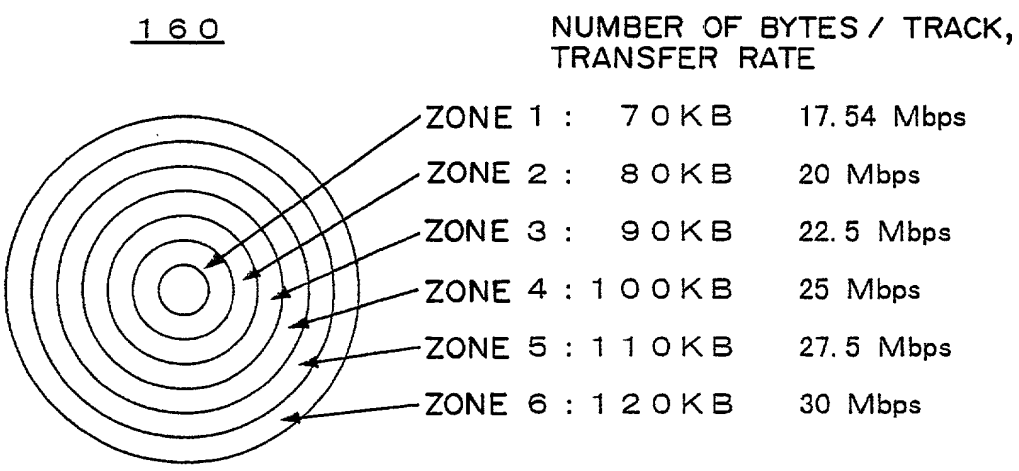

160
NUMBER OF BYTES / TRACK,
TRANSFER RATE
ZONE 1 : 70KB 17.54 Mbps
ZONE 2 : 80KB 20 Mbps
ZONE 3 : 90KB 22.5 Mbps
ZONE 4 : 100KB 25 Mbps
ZONE 5 : 110KB 27.5 Mbps
ZONE 6 : 120KB 30 Mbps

FIG. 52

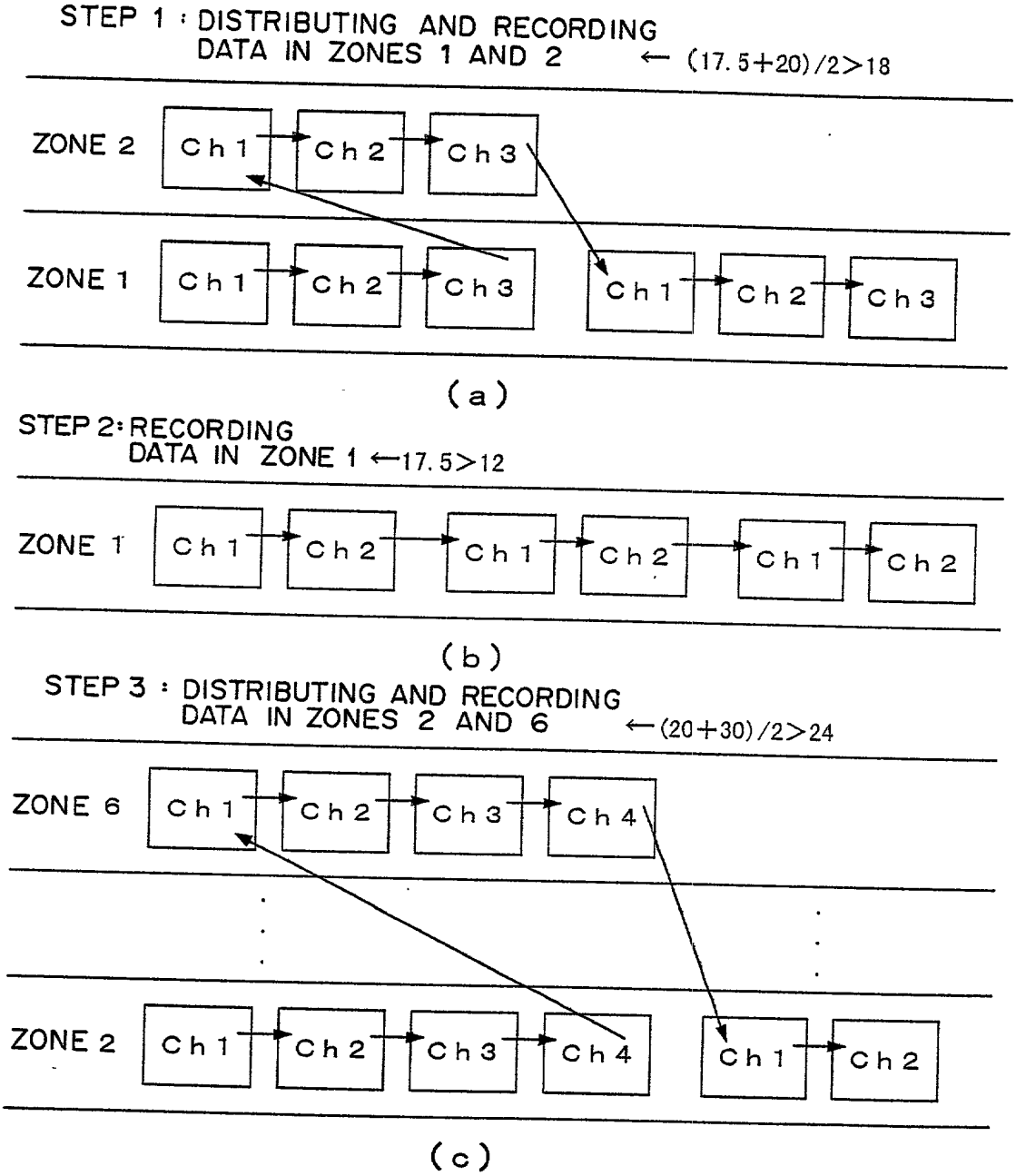

STEP 1 : DISTRIBUTING AND RECORDING DATA IN ZONES 1 AND 2
← (17.5+20)/2>18
ZONE 2
Ch 1
Ch 2
Ch 3
ZONE 1
Ch 1
Ch 2
Ch 3
Ch 1
Ch 2
Ch 3
(a)
STEP 2: RECORDING DATA IN ZONE 1 ←17.5>12
ZONE 1
Ch 1
Ch 2
Ch 1
Ch 2
Ch 1
Ch 2
(b)
STEP 3 : DISTRIBUTING AND RECORDING DATA IN ZONES 2 AND 6
←(20+30)/2>24
ZONE 6
Ch 1
Ch 2
Ch 3
Ch 4
ZONE 2
Ch 1
Ch 2
Ch 3
Ch 4
Ch 1
Ch 2
(c)

FIG. 53

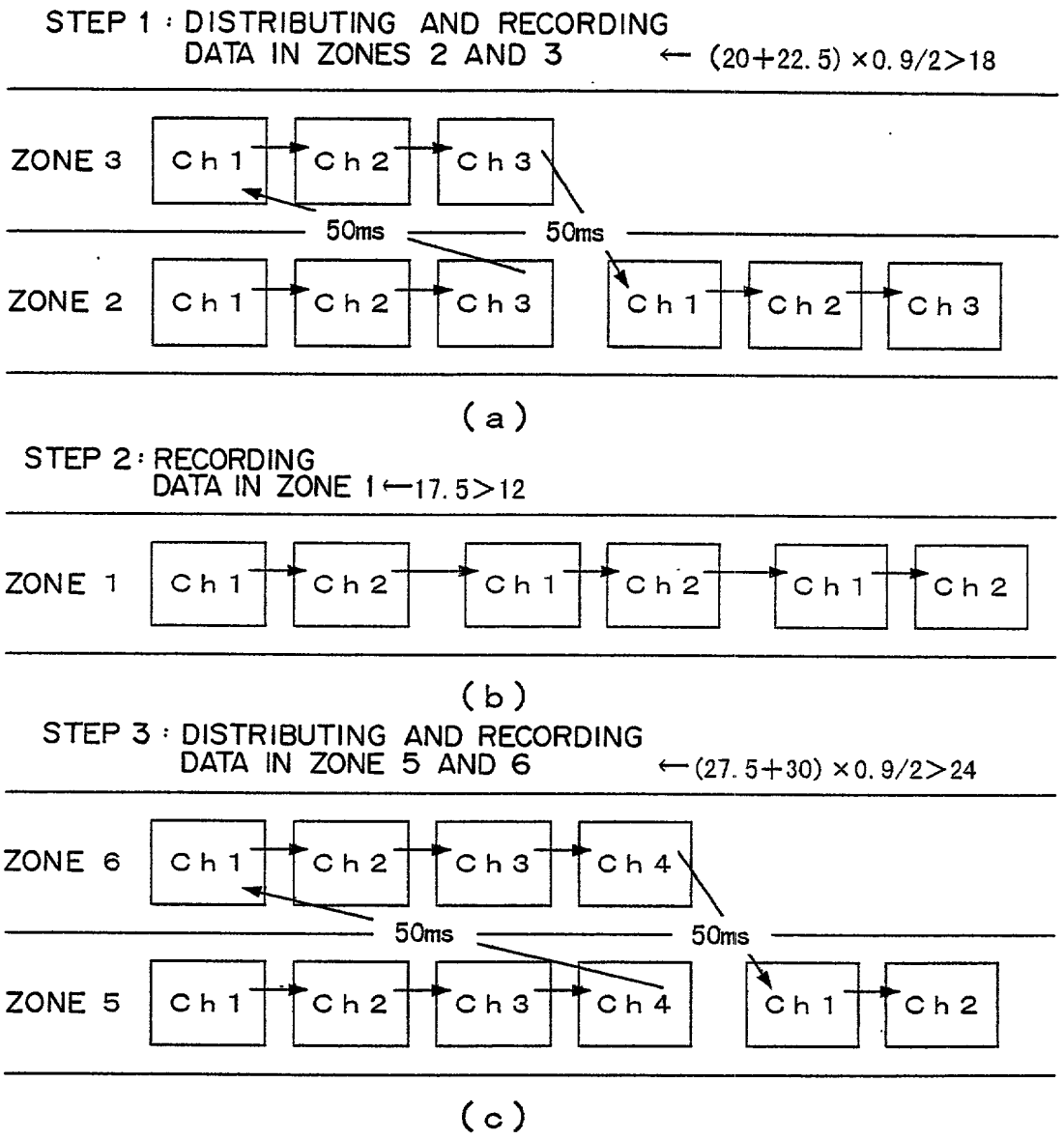

STEP 1 : DISTRIBUTING AND RECORDING DATA IN ZONES 2 AND 3
← (20+22.5) ×0.9/2>18
ZONE 3
Ch 1
Ch 2
Ch 3
50ms
50ms
ZONE 2
Ch 1
Ch 2
Ch 3
Ch 1
Ch 2
Ch 3
(a)
STEP 2 : RECORDING DATA IN ZONE 1 ←17.5>12
ZONE 1
Ch 1
Ch 2
Ch 1
Ch 2
Ch 1
Ch 2
(b)
STEP 3 : DISTRIBUTING AND RECORDING DATA IN ZONE 5 AND 6
←(27.5+30) ×0.9/2>24
ZONE 6
Ch 1
Ch 2
Ch 3
Ch 4
50ms
50ms
ZONE 5
Ch 1
Ch 2
Ch 3
Ch 4
Ch 1
Ch 2
(c)

FIG. 54

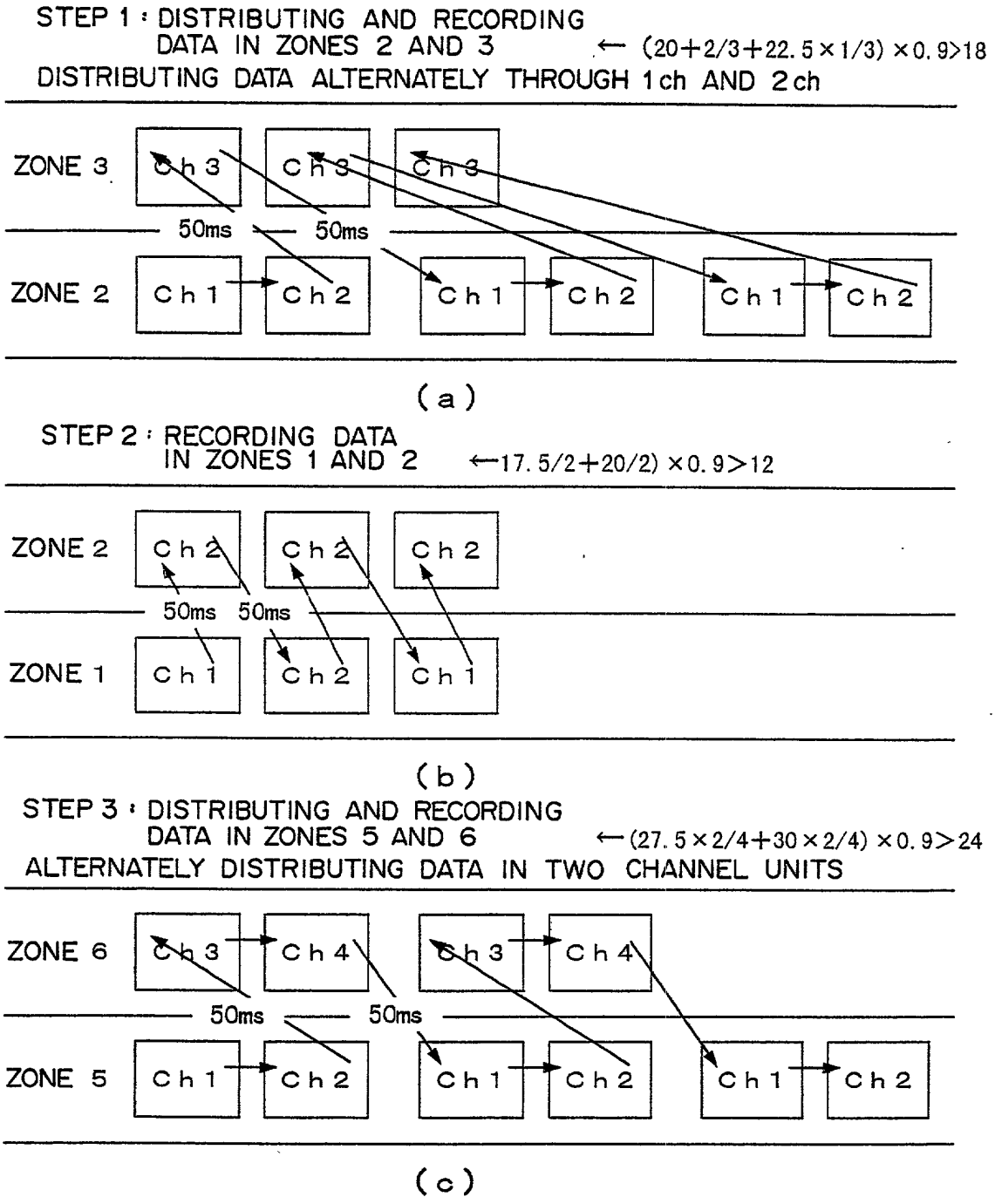

STEP 1 : DISTRIBUTING AND RECORDING DATA IN ZONES 2 AND 3
← (20+2/3+22.5×1/3) ×0.9>18
DISTRIBUTING DATA ALTERNATELY THROUGH 1ch AND 2ch
ZONE 3
Ch 3
Ch 3
Ch 3
50ms
50ms
ZONE 2
Ch 1
Ch 2
Ch 1
Ch 2
Ch 1
Ch 2
(a)
STEP 2 : RECORDING DATA IN ZONES 1 AND 2
←17.5/2+20/2) ×0.9>12
ZONE 2
Ch 2
Ch 2
Ch 2
50ms
50ms
ZONE 1
Ch 1
Ch 2
Ch 1
(b)
STEP 3 : DISTRIBUTING AND RECORDING DATA IN ZONES 5 AND 6
←(27.5×2/4+30×2/4) ×0.9>24
ALTERNATELY DISTRIBUTING DATA IN TWO CHANNEL UNITS
ZONE 6
Ch 3
Ch 4
Ch 3
Ch 4
50ms
50ms
ZONE 5
Ch 1
Ch 2
Ch 1
Ch 2
Ch 1
Ch 2
(c)

FIG. 55

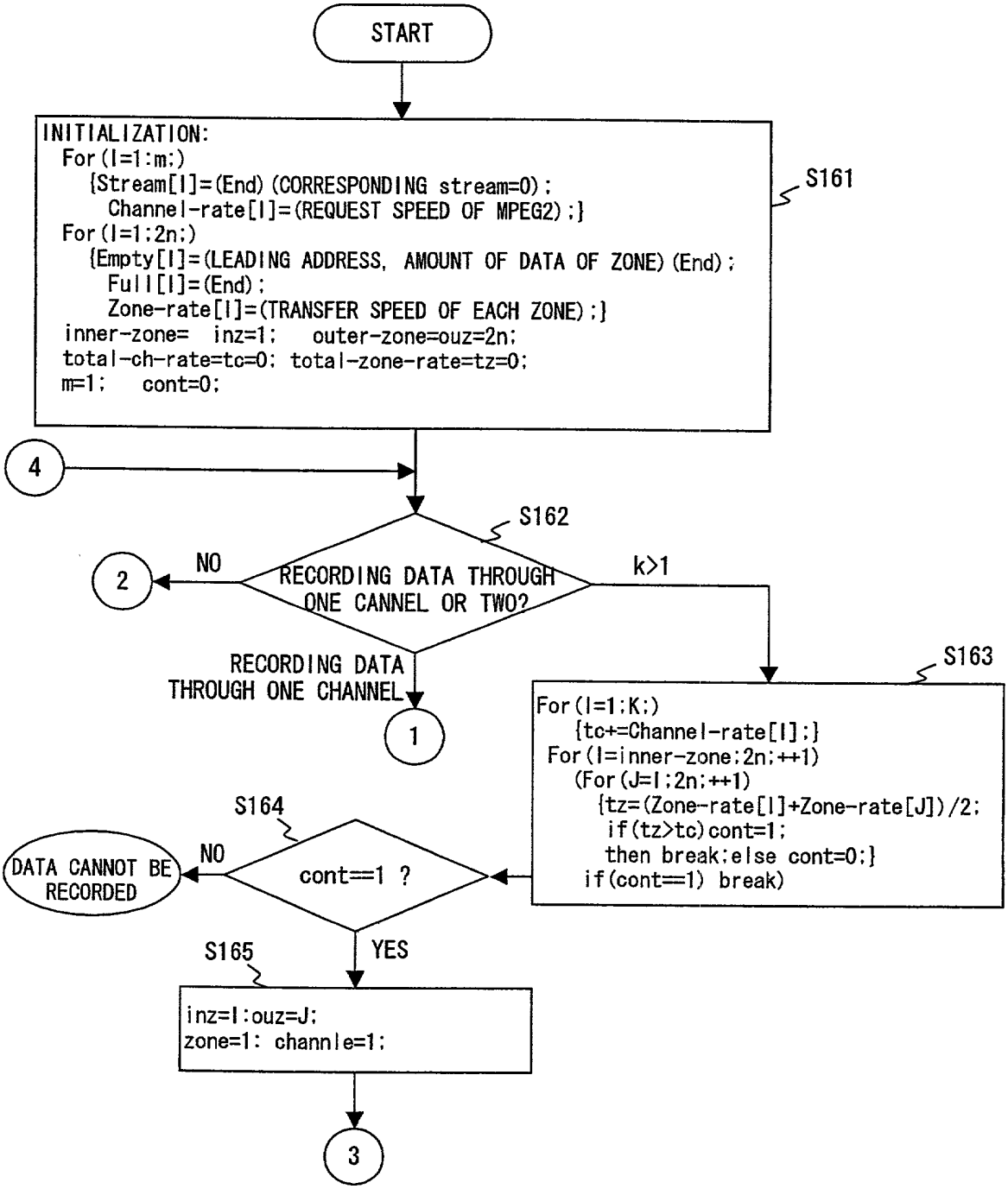

START
INITIALIZATION:
For(I=1:m;)
{Stream[I]=(End)(CORRESPONDING stream=0);
Channel-rate[I]=(REQUEST SPEED OF MPEG2);}
For(I=1;2n;)
{Empty[I]=(LEADING ADDRESS, AMOUNT OF DATA OF ZONE)(End);
Full[I]=(End);
Zone-rate[I]=(TRANSFER SPEED OF EACH ZONE);}
inner-zone= inz=1; outer-zone=ouz=2n;
total-ch-rate=tc=0; total-zone-rate=tz=0;
m=1; cont=0;
S161
4
S162
RECORDING DATA THROUGH ONE CANNEL OR TWO?
NO
2
k>1
RECORDING DATA THROUGH ONE CHANNEL
1
S163
For(I=1;K;)
{tc+=Channel-rate[I];}
For(I=inner-zone;2n;++1)
(For(J=I;2n;++1)
{tz=(Zone-rate[I]+Zone-rate[J])/2;
if(tz>tc)cont=1;
then break;else cont=0;}
if(cont==1) break)
S164
cont==1 ?
NO
DATA CANNOT BE RECORDED
YES
S165
inz=I:ouz=J;
zone=1: channle=1;
3

FIG. 56

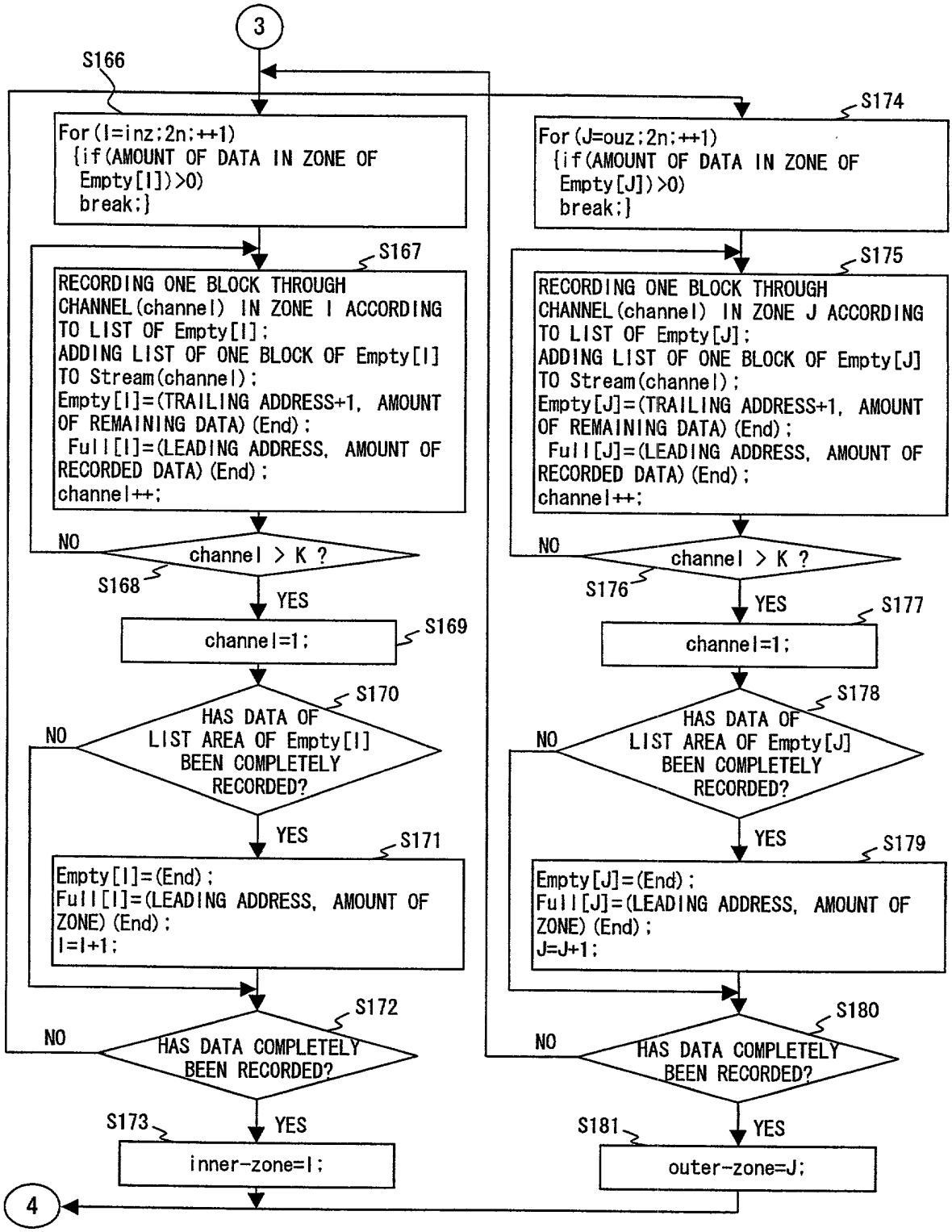

3
S166
For(I=inz;2n;++1)
{if(AMOUNT OF DATA IN ZONE OF
Empty[I])>0)
break;}
S167
RECORDING ONE BLOCK THROUGH
CHANNEL(channel) IN ZONE I ACCORDING
TO LIST OF Empty[I];
ADDING LIST OF ONE BLOCK OF Empty[I]
TO Stream(channel);
Empty[I]=(TRAILING ADDRESS+1, AMOUNT
OF REMAINING DATA)(End);
Full[I]=(LEADING ADDRESS, AMOUNT OF
RECORDED DATA)(End);
channel++;
NO
channel > K ?
S168
YES
channel=1;
S169
S170
HAS DATA OF
LIST AREA OF Empty[I]
BEEN COMPLETELY
RECORDED?
NO
YES
S171
Empty[I]=(End);
Full[I]=(LEADING ADDRESS, AMOUNT OF
ZONE)(End);
I=I+1;
S172
HAS DATA COMPLETELY
BEEN RECORDED?
NO
YES
S173
inner-zone=I;
4
S174
For(J=ouz;2n;++1)
{if(AMOUNT OF DATA IN ZONE OF
Empty[J])>0)
break;}
S175
RECORDING ONE BLOCK THROUGH
CHANNEL(channel) IN ZONE J ACCORDING
TO LIST OF Empty[J];
ADDING LIST OF ONE BLOCK OF Empty[J]
TO Stream(channel);
Empty[J]=(TRAILING ADDRESS+1, AMOUNT
OF REMAINING DATA)(End);
Full[J]=(LEADING ADDRESS, AMOUNT OF
RECORDED DATA)(End);
channel++;
NO
channel > K ?
S176
YES
S177
channel=1;
S178
HAS DATA OF
LIST AREA OF Empty[J]
BEEN COMPLETELY
RECORDED?
NO
YES
S179
Empty[J]=(End);
Full[J]=(LEADING ADDRESS, AMOUNT OF
ZONE)(End);
J=J+1;
S180
HAS DATA COMPLETELY
BEEN RECORDED?
NO
YES
S181
outer-zone=J;

FIG. 57

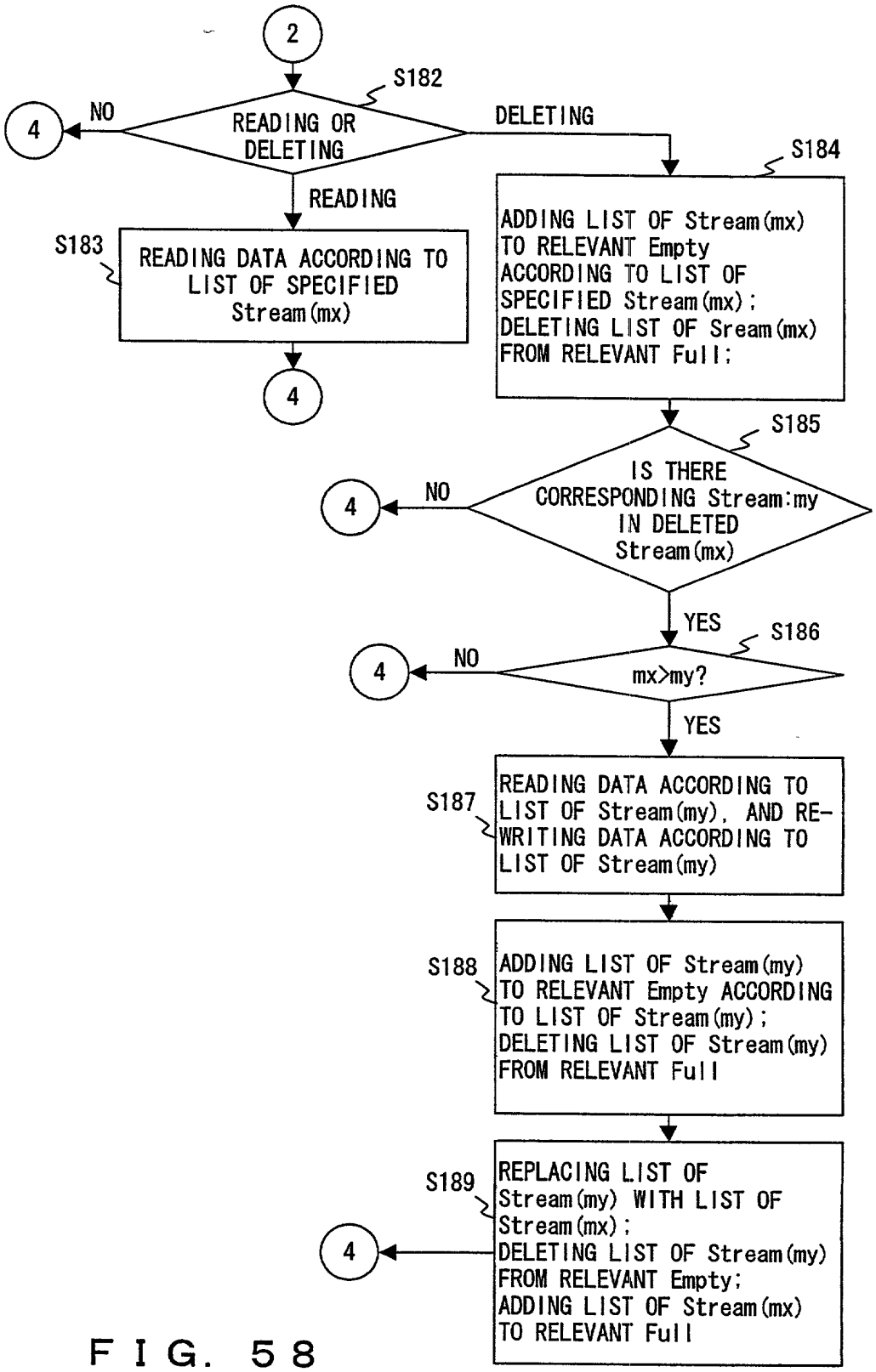

2
S182
READING OR DELETING
NO
4
DELETING
READING
S183
READING DATA ACCORDING TO LIST OF SPECIFIED Stream(mx)
4
S184
ADDING LIST OF Stream(mx) TO RELEVANT Empty ACCORDING TO LIST OF SPECIFIED Stream(mx); DELETING LIST OF Sream(mx) FROM RELEVANT Full;
S185
IS THERE CORRESPONDING Stream:my IN DELETED Stream(mx)
NO
4
YES
S186
mx>my?
NO
4
YES
S187
READING DATA ACCORDING TO LIST OF Stream(my), AND RE-WRITING DATA ACCORDING TO LIST OF Stream(my)
S188
ADDING LIST OF Stream(my) TO RELEVANT Empty ACCORDING TO LIST OF Stream(my); DELETING LIST OF Stream(my) FROM RELEVANT Full
S189
REPLACING LIST OF Stream(my) WITH LIST OF Stream(mx); DELETING LIST OF Stream(my) FROM RELEVANT Empty; ADDING LIST OF Stream(mx) TO RELEVANT Full
4

FIG. 58

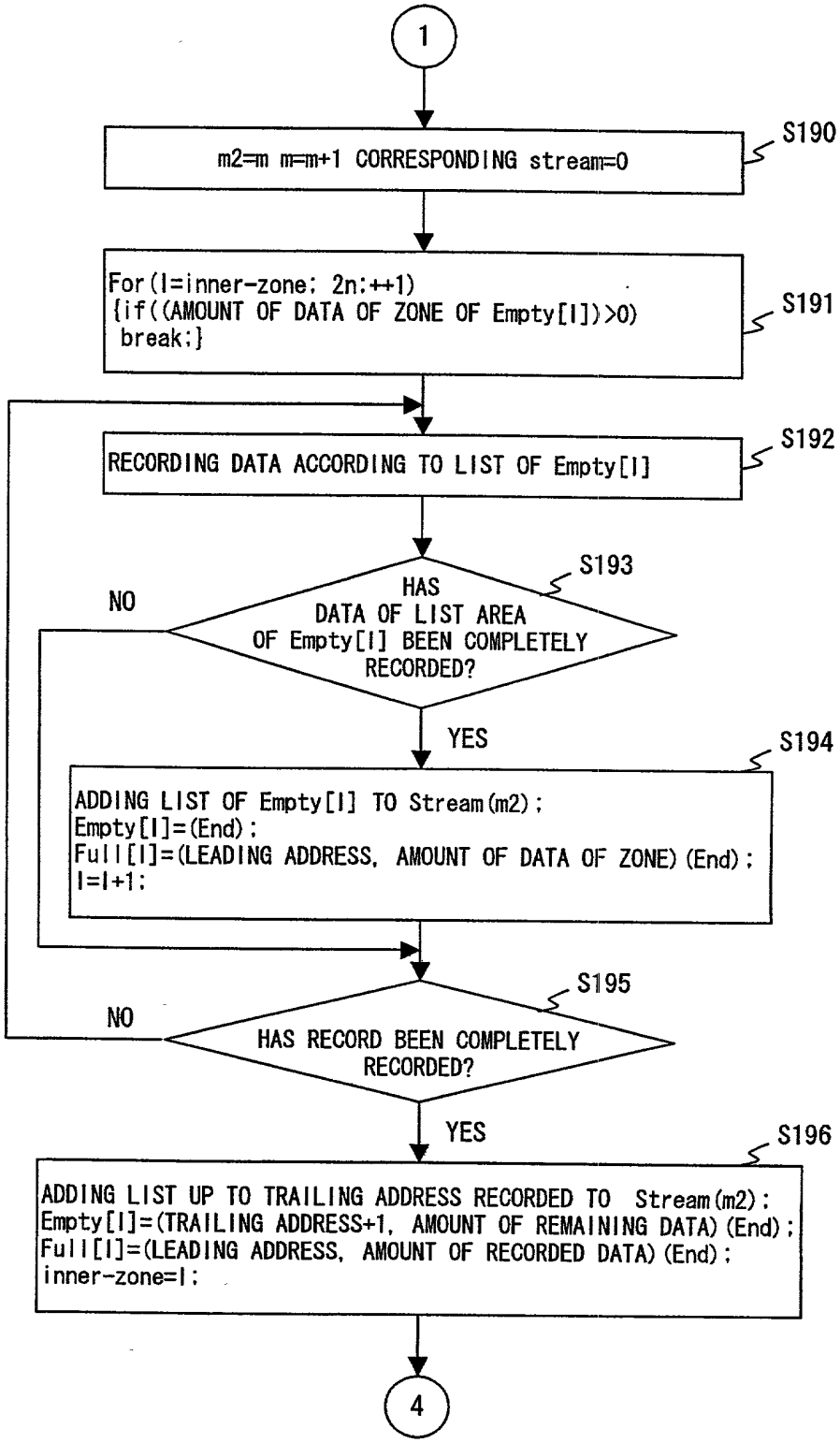

1
m2=m m=m+1 CORRESPONDING stream=0
S190
For(I=inner-zone; 2n;++1)
{if((AMOUNT OF DATA OF ZONE OF Empty[I])>0)
break;}
S191
RECORDING DATA ACCORDING TO LIST OF Empty[I]
S192
S193
HAS DATA OF LIST AREA OF Empty[I] BEEN COMPLETELY RECORDED?
NO
YES
S194
ADDING LIST OF Empty[I] TO Stream(m2);
Empty[I]=(End);
Full[I]=(LEADING ADDRESS, AMOUNT OF DATA OF ZONE)(End);
I=I+1;
S195
HAS RECORD BEEN COMPLETELY RECORDED?
NO
YES
S196
ADDING LIST UP TO TRAILING ADDRESS RECORDED TO Stream(m2);
Empty[I]=(TRAILING ADDRESS+1, AMOUNT OF REMAINING DATA)(End);
Full[I]=(LEADING ADDRESS, AMOUNT OF RECORDED DATA)(End);
inner-zone=I;
4

FIG. 59

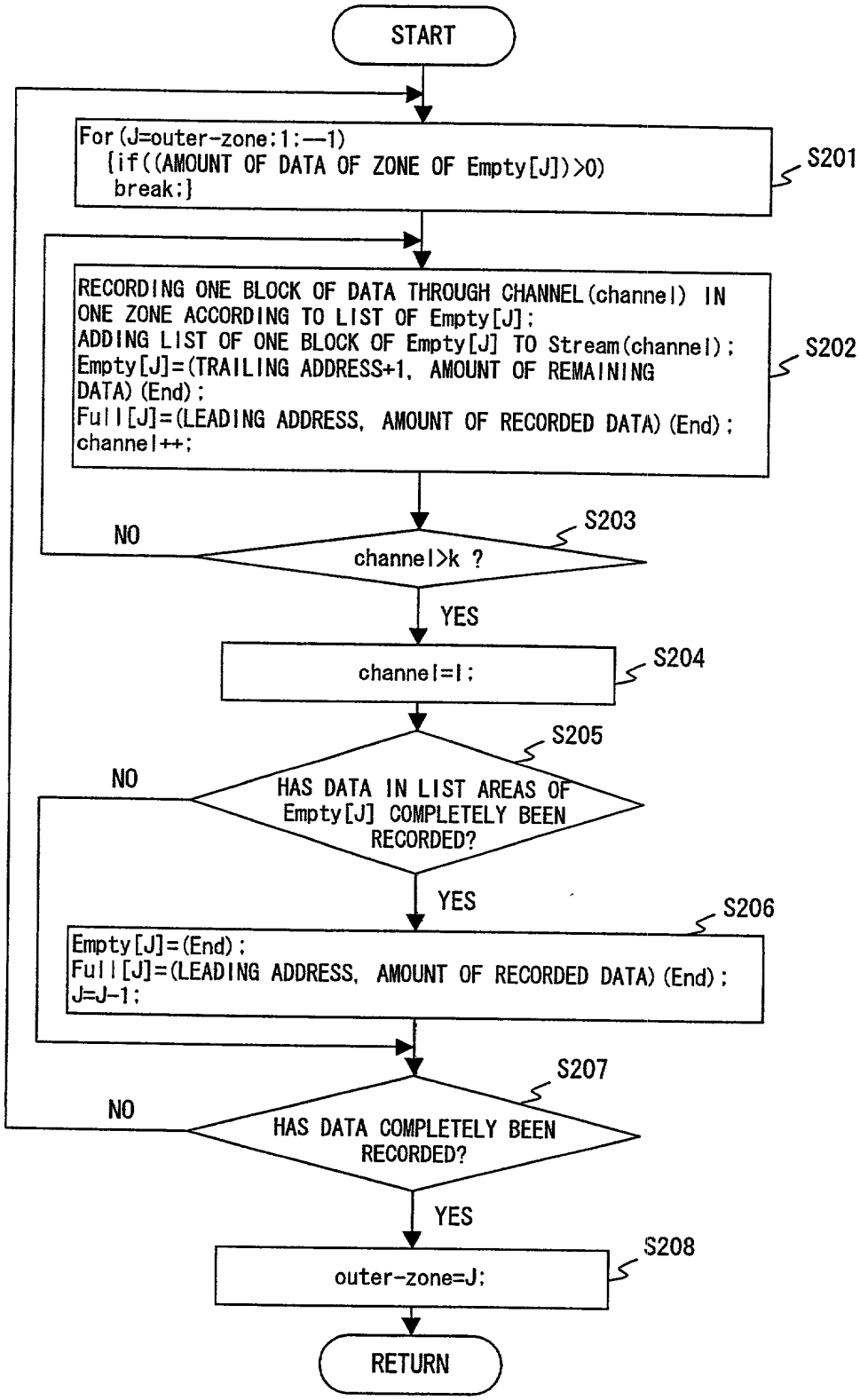

START
For(J=outer-zone:1:—1)
{if((AMOUNT OF DATA OF ZONE OF Empty[J])>0)
break;}
S201
RECORDING ONE BLOCK OF DATA THROUGH CHANNEL(channel) IN ONE ZONE ACCORDING TO LIST OF Empty[J];
ADDING LIST OF ONE BLOCK OF Empty[J] TO Stream(channel);
Empty[J]=(TRAILING ADDRESS+1, AMOUNT OF REMAINING DATA)(End);
Full[J]=(LEADING ADDRESS, AMOUNT OF RECORDED DATA)(End);
channel++;
S202
NO
channel>k ?
S203
YES
channel=1;
S204
NO
HAS DATA IN LIST AREAS OF Empty[J] COMPLETELY BEEN RECORDED?
S205
YES
Empty[J]=(End);
Full[J]=(LEADING ADDRESS, AMOUNT OF RECORDED DATA)(End);
J=J-1;
S206
NO
HAS DATA COMPLETELY BEEN RECORDED?
S207
YES
outer-zone=J;
S208
RETURN

FIG. 60